

# For Reference

---

**NOT TO BE TAKEN FROM THIS ROOM**

# For Reference

---

NOT TO BE TAKEN FROM THIS ROOM

Ex LIBRIS  
UNIVERSITATIS  
ALBERTAENSIS









THE UNIVERSITY OF ALBERTA

A GENERALIZED MATERIAL AND ENERGY BALANCE

COMPUTER PROGRAM

BY

RONALD E. NIEMAN

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

IN

CHEMICAL ENGINEERING

FACULTY OF ENGINEERING

DEPARTMENT OF CHEMICAL AND PETROLEUM ENGINEERING

EDMONTON, ALBERTA

APRIL, 1967



UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled A GENERALIZED MATERIAL AND ENERGY BALANCE COMPUTER PROGRAM submitted by Ronald E. Nieman, B.Sc., in partial fulfilment of the requirements for the degree of Master of Science in Chemical Engineering.



## ABSTRACT

This thesis describes a computer program which performs material and/or energy balance calculations of an arbitrary process. It employs information specified in input streams to, and output streams from, a balance envelope containing one or more units. This material and energy balance program is capable of solving many problems when used with its own executive routine but it was integrated into the logic of an existing program, PACER, so as to extend the power of both programs.

Many of the executive programs reported in the literature, including PACER, were only capable of handling problems in which all feed streams were totally specified.

The combined program may simplify the solution of some problems by eliminating the recycle calculations which PACER would perform if used alone. This is illustrated in the example problems presented in the Appendices.

The rule of calculation recommended for stability is the calculation of physical outputs from known inputs. Through the solution of material and energy balances around feasible balance envelopes located by the balance routine logic, this rule of calculation is preserved in many instances.

Subroutines for the calculation of dew point and bubble point temperatures, qualities and enthalpies were included in the executive program to permit calculation of



## ACKNOWLEDGEMENTS

The author wishes to express sincere appreciation to Dr. D.G. Fisher for his supervision of this investigation. The staff of the Computing Center is gratefully acknowledged for processing the digital computer programs submitted.

Financial assistance supplied by the Chemcell Limited Fellowship and leave granted by the Royal Canadian Air Force are also gratefully acknowledged.





## TABLE OF CONTENTS

|  | <u>Page</u> |
|--|-------------|
| I. INTRODUCTION  | 1           |
| A. Executive Program Concept   | 3           |
| B. Executive Logic   | 5           |
| C. Material and Energy Balance Calculation                                     | 7           |
| II. REVIEW OF THE LITERATURE   | 10          |
| A. Input Phase   | 11          |
| B. Execution Phase   | 14          |
| 1. Direct Method   | 15          |
| 2. Successive Substitution   | 17          |
| 3. Combination of Approaches   | 19          |
| III. THE MATERIAL AND ENERGY BALANCE COMPUTER PROGRAM                          | 24          |
| A. Input Data for Integrated Program   | 24          |
| 1. Summary of Input Data Cards   | 24          |
| 2. Adapting the Flowsheet for Use on the Digital Computer                      | 30          |
| 3. Restrictions Imposed by PACER   | 35          |
| B. Integration of the Material and Energy Balance Routine into the PACER Logic | 36          |
| C. Logic of the Material and Energy Balance Routine                            | 44          |
| D. The Material and Energy Balance Computer Program                            | 55          |
| 1. Subroutine BALNCE   | 58          |
| 2. Subroutine SEQCAL   | 64          |
| 3. Subroutine SETEQ  | 67          |
| 4. Subroutine SIMCAL   | 72          |
| 5. Subroutine ENTHPY   | 73          |
| 6. Subroutine BUBPT  | 76          |
| 7. Subroutine DEWPT  | 78          |
| 8. Subroutine QUAL   | 78          |
| 9. Subroutine EQUILK   | 81          |



|                           | <u>Page</u> |
|---------------------------|-------------|
| IV. EQUIPMENT SUBROUTINES | 83          |
| V. FUTURE WORK            | 85          |
| VI. CONCLUSIONS           | 87          |
| VII. BIBLIOGRAPHY         | 90          |



## APPENDICES

|  | <u>Page</u> |
|--|-------------|
| APPENDIX A            Example I                | A1          |
| APPENDIX B            Example II               | B1          |
| APPENDIX C            Example III              | C1          |
| APPENDIX D            Example IV               | D1          |
| APPENDIX E            Running Times of Program | E1          |
| APPENDIX F            Material Balance Program | F1          |
| Nomenclature                                   | F2          |
| Subroutine Listings                            | F9          |



## LIST OF TABLES

| <u>Table<br/>Number</u> | <u>Caption</u>  | <u>Page</u> |
|-------------------------|---|-------------|
| I                       | Process Matrix for Process Shown in<br>Figure III     | 12          |
| II                      | Association Matrix for Process Shown<br>in Figure III | 12          |
| III                     | Naphtali's Matrix for Process Shown in<br>Figure III  | 12          |
| IV                      | Example Reactor Data Representation                   | 32          |





## LIST OF FIGURES

| <u>Figure<br/>Number</u> | <u>Caption</u>  | <u>Page</u> |
|--------------------------|---|-------------|
| I                        | A Process Flow Diagram  | 2           |
| II                       | Flow Diagram of a Unit  | 6           |
| III                      | Example Process Flowsheet   | 13          |
| IV                       | Representation of a Reactor on a<br>Process Flowsheet             | 31          |
| V                        | Representation of a Distillation Column<br>on a Process Flowsheet | 34          |
| VI                       | Flow Diagram Illustrating the Logic of<br>the Integrated Program  | 37          |
| VII                      | Flow Diagram of MAJOR Calling Sequence                            | 42          |
| VIII                     | Representation of Two Phase Flow                                  | 50          |
| IX                       | A Balance Envelope and Associated Streams                         | 54          |
| X                        | Flow Diagram of the Material and Energy<br>Balance Routine Logic  | 57          |
| XI                       | Flow Diagram for Subroutine BALNCE                                | 59          |
| XII                      | Flow Diagram for Subroutine SEQCAL                                | 66          |
| XIII                     | Flow Diagram for Subroutine SETEQ                                 | 69          |
| XIV                      | Flow Diagram for Subroutine ENTHPY                                | 74          |
| XV                       | Two Phase Flow in a Stream  | 79          |



## I. INTRODUCTION

A part of most chemical engineering problems, notably design problems, is the material and energy balance calculation of the process under consideration. This thesis deals with computer programs written to accomplish this task.

This tedious task of calculating steady state material and energy balances was previously done by hand calculation. Often the engineer was able to employ his intuition and experience in completing the balance of the process under consideration especially when recycle loops such as Units 1, 2, 4 and 5 in Figure I were involved. To perform the material and energy balance of a process manually the engineer required:

1. the process flow diagram (Figure I)
2. physical property and engineering data
3. a mathematical model for each unit in the process which completes the material and energy balance for that unit.

From this the experienced engineer attacked each problem in the same manner. He determined:

1. the sequence of calculation
2. if any recycle loops were present and if so how best to accomplish the trial and error calculations
3. a method of convergence (or in other words a method to decide what values to use for the calculation of the next iteration when recycle is involved) and a test for the existence of an "acceptable" solution.



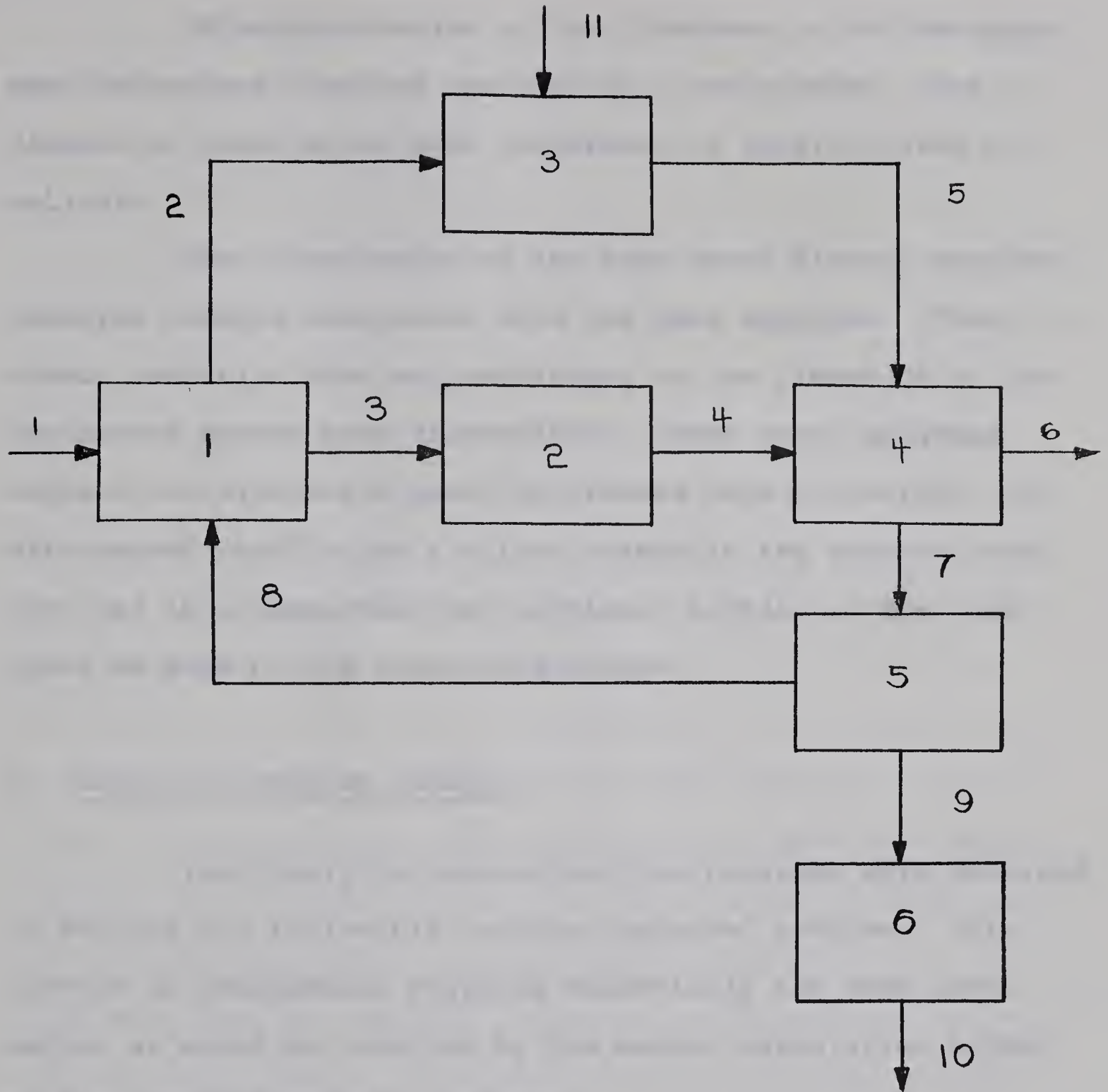


Figure I

A Process Flow Diagram





Effecting changes in the flowsheet or to the equipment parameters involved resulted in a new problem. The element of human error made comparison of related cases unreliable.

The introduction of the high speed digital computer rendered results consistent with the data supplied. Thus, trends resulting from any variations to the flowsheet or its parameters became more discernible. These early programs written to calculate a specific process were inflexible. As with manual calculation a slight change in the process often resulted in a completely new problem. Little, if any, use could be made of the previous solution.

#### A. Executive Program Concept

Inevitably the executive type programs were destined to replace the inflexible "special-purpose" program. This concept in programming requires essentially the same information as would be required by the manual calculation method or the "special-purpose" program:

1. flow diagram or flowsheet
2. engineering and physical property data
3. an equipment subroutine for each different unit in the process. The equipment subroutine effects the unit calculation. (It completes the material and energy balance around a unit and/or the equipment design.)





The executive system is essentially independent of chemical engineering and hence does not require the retraining of experienced chemical engineers for its development.

The executive system uses a logic in the approach to the problem of the material and energy balance calculation which is often identical to that outlined above for the manual method. The executive program makes the decisions which the engineer would have to make in solving the same problem by hand. However, the data and the process flowsheet must be encoded so as to be interpreted by the digital computer.

To effect changes in the process layout using this type of system requires only alteration to the input data and possibly the addition, or replacement, of equipment subroutines. Indeed, to change the problem completely only requires this alteration. The flexibility of this executive concept usually outweighs the efficiency associated with the "special-purpose" program. The executive program is at a disadvantage when the identical design or simulation is to be repeated many times. It is here that the "special-purpose" program is most economical.

Use of the executive program releases the experienced engineer from the laborious and repetitious task of completing material and energy balances to devote his time to more creative work.



## B. Executive Logic

The basic rule for calculation common to the executive programs presented in the literature is, knowing all the inputs to a particular unit, calculate the outputs from the equipment model. An equipment subroutine is supplied for each different equipment in the process to effect the unit calculations.

If this rule of calculation is strictly adhered to, greater numerical stability is the result in the overall simulation or design. Rinard(15) emphasizes this statement by the following example. By material balance around unit A in Figure II,

$$X(4) + X(6) = X(1) + X(2) + X(3) \quad (1)$$

where  $X(I)$  is an element of a vector  $\underline{X}(I)$  corresponding to a conserved component flow rate.  $\underline{X}(I)$  is simply a vector of flow rates of the components in stream I. It may also include the enthalpy of the stream as one additional element.

The unit calculation supplies an additional relationship,

$$X(4) = KA_1 X(1) + KA_2 X(2) + KA_3 X(3) \quad (2)$$

where  $KA_i$  is the fraction of the component flow rate of stream  $i$  entering unit A which leaves in stream 4.

$$0 \leq KA_i \leq 1 \quad \text{for} \quad i = 1, 2, 3$$



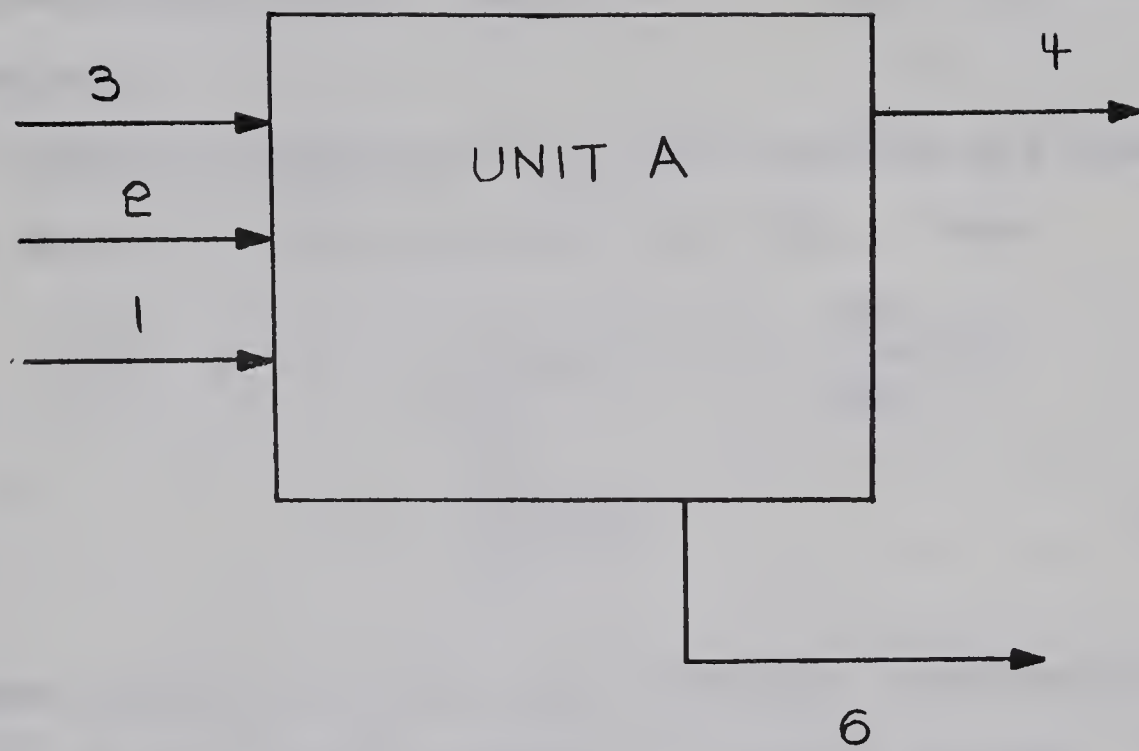


Figure II  
Flow Diagram of a Unit





Knowing the input variables  $X(1)$ ,  $X(2)$  and  $X(3)$  the output streams  $X(4)$  and  $X(6)$  can be calculated. For example, combining equations (1) and (2):

$$X(6) = (1-KA_1)X(1) + (1-KA_2)X(2) + (1-KA_3)X(3) \quad (3)$$

The input-to-output gains  $(1-KA_i)$  for  $i = 1, 2, 3$  are bounded by zero and one.

However, supposing  $X(1)$ ,  $X(3)$  and  $X(4)$  are known and it is desired to calculate  $X(2)$  and  $X(6)$ . Then:

$$\begin{aligned} X(6) = & -\left(1 - \frac{1}{KA_2}\right)X(4) + \left(1 - \frac{KA_1}{KA_2}\right)X(1) \\ & + \left(1 - \frac{KA_3}{KA_2}\right)X(3) \end{aligned} \quad (4)$$

This renders input-to-output gains unbounded accentuating errors made in the initial guesses of recycle streams. Therefore, where possible it is advisable to maintain the rule of calculation - that is, calculation of outputs from inputs.

### C. Material and Energy Balance Calculation

It is to preserve this rule of calculation that a material and energy balance routine was introduced to an executive program. Using the executive program, PACER(9), the executive logic is integrated with a material and energy balance computer program to allow the solution of some problems previously beyond the scope of PACER.





PACER was chosen since it was available in documented form in a Master of Science thesis(9). Later versions of PACER have since been developed(21). Any other executive program using the same sequential method of calculation as PACER could be improved with the addition of the subroutines of this material and energy balance program. PACER employs the trial and error method of calculation identical to that which would be used by an engineer doing the material and energy balance manually.

PACER was constructed to handle problems that had all fresh feed streams (streams 1 and 11 in Figure I) to the process specified. Thus, since all inputs to the system are given, the basic rule of calculation is preserved and all streams can be calculated given values of parameters involved and equipment subroutines. Often the problem is slightly altered and some fresh feed stream variables are not specified whereas some quantities are specified in inter-equipment streams (streams 2, 3, 4, 5, 7, 8 and 9 in Figure I) or product streams (streams 6 and 10 in Figure I). PACER could not handle this case. This work describes the subroutines which when added to PACER enable the integrated program to handle some of these problems.

Given some information other than feed specifications the modified executive system attempts material and/or energy balances around units yielding a plausible balance



envelope in order to calculate unknown inputs. Even when all inputs to the process are specified an attempt is made by material and energy balance subroutines to exploit any information specified in product or inter-equipment streams.

The material and energy balance subroutines integrated with PACER are also available for balance calculations around a single unit. Therefore, the equipment subroutines need not contain material and energy balance calculations as such.

The addition of the enthalpy balance subroutine to PACER required subroutines to calculate dew points, bubble points, qualities, and equilibrium ratios which are also available to the user.





## II. REVIEW OF THE LITERATURE

In the chemical engineering field such executive programs as Kellogg's "Flexible Flowsheet" (8), Shell's "Cheops", IBM's "Generalized Interrelated Flow Simulation" (GIFS), PACER (9) and SPEEDUP (21) have been developed. The literature contains many papers devoted to the presentation of some of these and other similar programs. Some similarities and differences associated with these executive programs for the calculation of steady state material and energy balances are presented in subsequent paragraphs. The general structure around which each of these programs is built is three-phase:

1. an executive system including execution, input and output subroutines
2. equipment subroutines to effect unit calculations
3. engineering and physical property data

It is this first phase of the executive program which is so widely discussed in the literature. Therefore, some time will be devoted to discussion of this phase in more detail. In order that the problem can be readily interpreted by the digital computer, the flowsheet, engineering data, and physical property data must be encoded numerically. A computational scheme must be employed and recycle loops identified. The following discussion presents several alternate methods proposed for these requirements.



### A. Input Subroutine

Common to each executive system is some form of encoding the flow diagram. "SPEEDUP"(21) uses a routine which has gone one step further than most systems and incorporated an interpreter into the data input routine. Thus, the data encoding the flow diagram takes a form familiar to the chemical engineer who is inexperienced with an executive system. As an example, the following is a valid statement of input data for "SPEEDUP":

INPUT 2 OF HEAT EXCHANGER 14 IS OUTPUT 4  
OF PLATE COLUMN 3.

This type of input format is then numerically encoded internally in a compact form similar to that used by PACER.

PACER and its later versions(9,22,23) use a process matrix to convey the information normally contained in a flow diagram. Using the process flow diagram shown in Figure III, the process matrix is shown in Table I.

The first element in a row is the equipment number followed by the equipment subroutine name. The remaining elements comprise a list of input streams to this particular unit (positive) followed by output streams (negative) from it.

In lieu of this process matrix Ravicz and Norman(14) use an association matrix,  $\underline{A}$ , where  $(a_{ij}) = 1$  if a stream joins module  $i$  to module  $j$  - otherwise  $(a_{ij}) = 0$ . This results





TABLE I

Process Matrix for Process Shown in Figure III

|   |        |   |   |   |   |   |   |   |
|---|--------|---|---|---|---|---|---|---|
| 1 | UNAME1 | 1 | 2 | - | 3 |   |   |   |
| 2 | UNAME2 | 3 | - | 2 | - | 4 | - | 5 |
| 3 | UNAME5 | 5 | - | 6 |   |   |   |   |

TABLE II

Association Matrix for Process Shown in Figure III

| i \ j | 1 | 2 | 3 |
|-------|---|---|---|
| 1     | 0 | 1 | 0 |
| 2     | 1 | 0 | 1 |
| 3     | 0 | 0 | 0 |

TABLE III

Naphtali's Matrix for Process Shown in Figure III

| i \ j | 1 | 2  | 3  | 4  | 5  | 6  |
|-------|---|----|----|----|----|----|
| 1     | 1 | 1  | -1 | 0  | 0  | 0  |
| 2     | 0 | -1 | 1  | -1 | -1 | 0  |
| 3     | 0 | 0  | 0  | 0  | 1  | -1 |



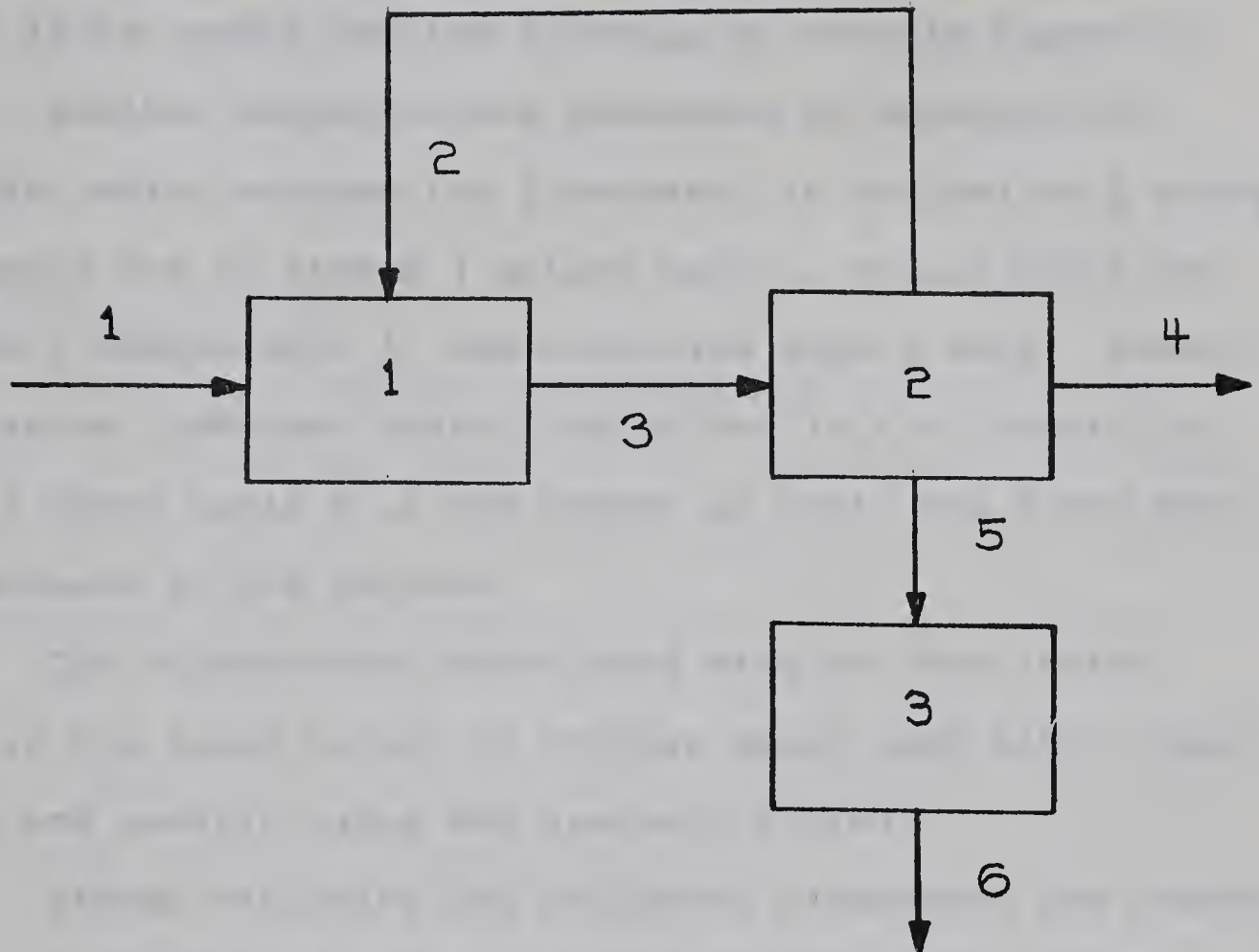


Figure III

Example Process Flowsheet



in an  $(n \times n)$  matrix where 'n' is the number of modules or units. Using the same flowsheet as an example (see Figure III) the association matrix is shown in Table II. This association matrix does not include feed or product stream information. However, it is useful for the location of recycle loops(13).

Another variation was presented by Naphtali(12). His matrix, which encodes the flowsheet, is defined as  $\underline{A}$  where  $(a_{ij})$  equals one if stream  $j$  enters unit  $i$ , equals minus one if stream  $j$  leaves unit  $i$ , and otherwise equals zero. Encoding the above flowsheet again, yields the  $(n \times m)$  matrix in Table III where again  $n$  is the number of units and  $m$  the number of streams in the process.

The disadvantage associated with the two latter methods is the large amount of storage space used since these matrices are usually large and sparsely filled.

Stream variables and equipment parameters are stored in assigned matrices in the executive program data system. For example, PACER stores information of a specific stream in a matrix row thus the process flowsheet of Figure III would require a matrix of six rows and a number of columns depending on the number of components involved.

## B. Execution Routine

There are three computational schemes used to approach the calculation of steady state material and energy balances of a process with arbitrary recycle:





1. direct or unit method(2,10,11,12,17,25,26)
2. sequential calculations assuming unknown inputs(3,8,9,15,18,21,22,23)(identical to the manual calculation discussed previously)
3. sequential calculation until a recycle stream is encountered, then calculation of the nests by direct method(13,14)

### 1. Direct Method

The direct method is employed by Rosen(17) using the method initiated by Nagiev(10,11) for handling processes with arbitrary recycle. It is necessary to know the component output of each unit as a general function of the component input.

Using the Nagiev method, the material balance for the  $i^{\text{th}}$  unit and the  $k^{\text{th}}$  component is(17):

$$-\alpha_{i1k}\lambda_{1k} - \alpha_{i2k}\lambda_{2k} - \dots + (1 - \alpha_{iik})\lambda_{ik} - \dots - \alpha_{ink}\lambda_{nk} = g_{iok} \quad (5)$$

$$k = 1, 2, \dots, m \quad (m \text{ components})$$

$$i = 1, 2, \dots, n \quad (n \text{ units})$$

where

$g_{iok}$  - fresh feed of component  $k$  to unit  $i$

$\alpha_{ijk}$  - weight fraction of component  $k$  entering unit  $j$  which is recycled to unit  $i$  (recycle fraction)





$\lambda_{ik}$  - total flow of component k to unit i from  
all n units

$g_{ijk}$  - weight flow of component k going to unit i  
from unit j ( $\lambda_{ik} = \sum_{j=0}^n g_{ijk}$ )

The recycle fractions ( $\alpha_{ijk}$ 's) are functions of process conditions and other components in the system. However, if the recycle fractions ( $\alpha_{ijk}$ 's) are constant, the component flow rates ( $\lambda_{ik}$ 's) can be calculated directly from the system of linear algebraic equations (5) given the fresh feeds ( $g_{iok}$ 's).

But in multicomponent separations, for an example, any one component may appear in several streams (or phases) flowing from the same unit. Letting symbols  $r_{ik}$ ,  $s_{ik}$ , ...  $q_{ik}$  and  $\gamma_{ji}$ ,  $\beta_{ji}$ , ...  $w_{ji}$  be these stream flows and phase recycle fractions respectively, the recycle fraction for component k to unit j from unit i is:

$$\alpha_{jik} = \left(\frac{r_{ik}}{\lambda_{ik}}\right) \gamma_{ji} + \left(\frac{s_{ik}}{\lambda_{ik}}\right) \beta_{ji} + \dots + \left(\frac{q_{ik}}{\lambda_{ik}}\right) w_{ji} \quad (6)$$

The split fractions ( $\frac{r_{ik}}{\lambda_{ik}}$ , ... ,  $\frac{q_{ik}}{\lambda_{ik}}$ ) for the p streams must be determined from the nature of the unit and process. For example, if unit i were a distillation column  $r_{ik}$  might be distillate and  $s_{ik}$  the bottom product flow rates for component k.



The phase recycle fractions are usually fixed and, given the fresh feeds to the process, the unknown split fractions are assumed ( $p \times n \times m$  guesses) and the recycle fractions are calculated from equation (6). Now it is a simple matter to solve the  $n$  simultaneous linear algebraic equations for the component feeds to each unit ( $\lambda_{ik}$ ) from equations (5). The split fractions can now be calculated from the unit calculation. An iterative technique yields a solution. However, this method does not take advantage of any structure of the network that might simplify the calculations.

## 2. Successive Substitution

Another approach is the classical chemical engineering approach, that of successive substitution. The method is to proceed sequentially calculating outputs from known, or previously calculated inputs. Unknown inputs are assumed known and their subsequent calculated values are used for the next iteration. This procedure is repeated until the two most recent values of all assumed stream variables are equal within some predetermined error. (This is the method used when manually calculating material and energy balances for a process.) Herein lies one of the pitfalls of this simple method: often convergence, the path along which the assumed variables approach the solution, is asymptotic and, although the assumed variables appear to have converged, actual convergence is only





very slow. Cavette(2) has published a comprehensive paper on the numerical methods of solution of material and energy balances for recycle processes. In this review convergence accelerators for this method are discussed. However, convergence can be hindered by such techniques. For example, when the interaction effects between components are large, these convergence "accelerators" may actually impede progress towards an acceptable solution.

The method of successive substitution also involves a sequence of calculation. An attempt is made by some authors(1,18,19,20,21) to select a sequence of calculation which minimizes the number of parameters which must be assumed by selecting an "optimum" sequence of calculation. Sargent(21) employs a dynamic programming algorithm in "SPEEDUP". On the other hand, Rubin(18) provides an algorithm based on systematic interchanges of units from an arbitrary sequence of units. The choice of the units to be interchanged at any stage is governed by the properties of a special matrix. Unfortunately only local optima are obtained by this method.

This procedure of finding the path of calculation yielding minimum number of recycle parameters to be assumed is not necessarily the optimum approach. Sargent(21) expresses the intention of extending his method to optimize a weighted sum of the number of recycle parameters assumed for a particular sequence and the difficulty which these parameters contribute to convergence. The goal here being to



assume the parameters which are most insensitive to the error associated with the initial guess.

PACER makes no attempt to find an optimum sequence of calculation. The sequence of calculation is mainly determined by scanning in the order listed in the process matrix. The streams to be assumed during the trial and error calculations may be specified as a part of the input data or selected by the executive program.

### 3. Combination of Approaches

The third computational scheme is merely a combination of the two previous methods. Calculations are carried out by calculating physical outputs from known physical inputs for each unit. However, when a recycle loop is encountered, the nest is identified and calculated by a direct method. Thus, this scheme requires a method for locating recycle nests. The nests are calculated when all inputs to them are known.

Ravicz and Norman(14) employ this logic for calculating recycle processes. The nests are located by using an "association matrix",  $\underline{A}$ , mentioned in the discussion of the input data routine and a method for location of cycles on a directed line graph(13). The elements  $a_{ij}$  of  $\underline{A}^\lambda$  where  $\lambda$  is an integer power, equals the number of distinct paths involving  $(\lambda+1)$  nodes going from node  $i$  to node  $j$ . The square of the association matrix contains information about





paths involving three units. Recycle nests are located by the appearance of non-zero elements on the diagonal of  $\underline{A}^\lambda$ . Therefore, considering  $\underline{A}^2$  again information concerning recycle nests involving two units is contained in the non-zero elements on the diagonal of  $\underline{A}^2$ .

The nests are calculated by the Newton-Raphson technique for solving non-linear simultaneous equations. Although this requires a large amount of computation per iteration, convergence is rapid for the near linear case. The method is outlined below.

Defining  $\underline{X}$  as a vector of stream flow variables (m component flow rates and the stream enthalpy for each recycle stream, m being the number of components), the multi-loop problem may be formulated as finding the roots of the function  $\phi(X)$ .

$$\underline{Y} = \phi(X) = \underline{R} - \underline{X} \quad (7)$$

$\underline{R}$ , a vector of stream variables equal in length to  $\underline{X}$ , is the calculated value of the recycle stream variables using the vector  $\underline{X}$  as a base point. Knowing that at the solution  $\underline{Y} = 0$  the desired change in  $\underline{Y}$  required to effect this equality is

$$\underline{\Delta Y} = -\underline{Y} \quad (8)$$

The problem is now to find the  $\underline{\Delta X}$  that results in this desired  $\underline{\Delta Y}$ .



$$\Delta \underline{X}^{(k)} = \underline{A}^{-1} \Delta \underline{Y}^{(k)} \quad (9)$$

where  $\underline{A}$  is the matrix of partial derivatives required for this technique  $(\frac{\partial X_i}{\partial Y_j}, i = 1, 2, \dots, n+1, j = 1, 2, \dots, n+1)$ .  
 $n = m$  for single loop problem.

$$\underline{X}^{(k)} = \underline{X}^{(k)} + \Delta \underline{X}^{(k)} \quad (10)$$

(superscript k denotes the iteration)

Therefore, assuming a vector  $\underline{X}^{(0)}$  as a base point, the process is evaluated yielding a value of  $\underline{R}$

$$\phi(\underline{X}^{(k)}) = \underline{Y}^{(k)} = \underline{R}^{(k)} - \underline{X}^{(k)} \quad (11)$$

Making a small positive change in  $\underline{X}^{(0)}$ , say  $\underline{DX}^{(0)}$ , the process is re-evaluated yielding a value for  $\phi(\underline{X}^{(k)} + \underline{DX}^{(k)})$ . The matrix of partial derivatives is constructed by finite difference.

$$(a_{ij})^{(k)} = \frac{\phi(X_i^{(k)} + DX_i^{(k)}) - \phi(X_i^{(k)})}{DX_j^{(k)}} \quad (12)$$

for  $i = 1, \dots, n+1, \quad j = 1, \dots, n+1$

The set of simultaneous linear algebraic equations (9) must be solved for  $\Delta \underline{X}^{(k)}$ . The procedure is then repeated

$$\underline{X}^{(k+1)} = \underline{X}^{(k)} + \Delta \underline{X}^{(k)} \quad (13)$$

Cavette(2) discusses modifications to this by the introduction of a method of interpolation into this equation.





The difficulty associated with this scheme arises in the evaluation of the partial derivatives.

There is a possibility that the strategies outlined above may be combined. Since direct iteration or successive substitution has a larger radius of convergence than the Newton-Raphson technique, it could be used to improve an initial guess. Then the accuracy of the answer can be refined using the Newton-Raphson method. This scheme would be useful in problems which involve large recycling streams where convergence of the direct iteration method becomes worse as the solution is approached. Here the behavior becomes more linear and Newton-Raphson iteration would work well.

Cavette(2) reviews most of the numerical methods presented in the literature for solving recycle processes. It is the most comprehensive paper on the subject to date.

Rinard and Ripps(15) discuss a method of utilizing information given about streams other than fresh feed streams which could be applied to executive programs with logic similar to that of PACER. They advise against the use of equipment subroutines to "back-calculate" unknown input variables from specified output variables. But rather, through the use of "controller" subroutines, the feed stream variables are altered from their original values to yield desired values for specified variables in output streams. The material and energy balance subroutines developed for this investigation, as indicated previously compute unknown input stream variables,





if possible, by material and energy balance around appropriate units.



### III. THE MATERIAL AND ENERGY BALANCE

#### COMPUTER PROGRAM

The material and energy balance program seeks balance envelopes around one or more units in a process in an attempt to solve for unknown input variables by exploiting specified output variables. Supplied with its own executive system this routine could solve many problems alone. However, integration of this routine into the PACER logic enhances the power of both programs. In a subsequent section the integration of this material and energy balance logic into PACER is presented with reference to flow diagrams. However, it would be helpful at this point to outline in brief the format of the data storage and input. A brief discussion of the handling of reactors and distillation columns will produce an understanding of the interpretation of flowsheet data by the integrated program.

A detailed discussion of the material and energy balance will follow with specific reference to the subroutines involved in the location of the balance envelope and in the setup and solution of the resulting equations.

#### A. Input Data for Integrated Program

##### 1. Summary of Input Data Cards

Reference is made to Mosler's thesis(9) for a more detailed description of the input data and its format.



1. Title Card
2. Run Control Card - run number and various option flags  
(A new option flag has been added that is the variable NEB. If NEB is one, only material balance calculations are to be executed. If NEB is two, then the energy balance calculations are to be completed also. However, when NEB is three, the energy balance calculations are to be attempted neglecting pressure correction.)
3. Dimension Card - contains the dimensions of the matrices and vectors
4. Process Matrix Run Control Card
5. Process Matrix Vector - encodes the flow diagram
  1. KPM (NEX, 1) - PACER equipment number  
- NEX
  2. NAME (NEX) - equipment subroutine name (blank if only the material and energy balance routine is to be used)
  3. NEXEQN (NEX) - equipment number positive if the balance routine is to be called before the equipment subroutine; negative if the equipment subroutine is to be called immediately that all inputs to the equipment NEX are known
  4. KPM (NEX, N3) - associated stream numbers; input streams are given first (positive) and outputs follow (negative) (N3 = 2,3, ...10)







6. Equipment Parameters Control Card - the number of equipment parameters vectors to input
7. Equipment Parameters Vector
  1. EN (NEN, 1) - PACER equipment number  
- NEN
  2. EN (NEN, 2) - PACER external equipment flag
  3. EN (NEN, K3) - other pieces of information determined by the equipment subroutine  
(K3 = 3, ... , NELMAX)
8. Equipment Control Parameters Control Card - the number of equipment control parameters vectors to input.
9. Equipment Control Parameters Vector
  1. ENC (NENC, 1) - PACER equipment number  
- NENC
  2. ENC (NENC, K3) - other pieces of information determined by the equipment subroutine  
(K3 = 2, ... , NECLMX)
10. Stream Variables Control Card - the number of stream variables matrix vectors to input
11. Stream Variables Matrix Vector - contains flow variables
  1. SN (NSN, 1) - PACER stream number  
- NSN
  2. SN (NSN, 2) - PACER stream flag  
- 0 if stream is an inter-equipment stream  
- 1 if stream is a feed stream  
- 2 if stream is a product stream



3. SN (NSN, 3) - total mass flow rate
4. SN (NSN, K3) - the next NOCOMP entries are mass fractions of the components in the stream; NOCOMP is the number of components in the system  
(K3 = 4, ... , NOCOMP + 3)
5. SN (NSN, K3) - these NOCOMP entries are component-mass flow variables (K3 = NOCOMP + 4, ... , 2 x NOCOMP + 3)
6. SN (NSN, 2 x NOCOMP + 4) - temperature in degrees rankine in stream which is an actual process stream  
- 10000; if stream is a dummy feed or product stream to a reactor  
- heat flow if the stream is of zero mass and represents a heat flow
12. Stream Control Variable Control Card - the number of stream control variables vectors to input
13. Stream Control Variables Vector - if stream is not a dummy stream or heat flow
  1. SNC (NSNC, 1) - PACER stream number  
- NSNC
  2. SNC (NSNC, 2) - PACER stream flag



3. SNC (NSNC, 3) - pressure (psia)
4. SNC (NSNC, 4) - the number of components in the stream  
(NCOMP) (NCOMP  $\leq$  NOCOMP)
5. SNC (NSNC, 5) - quality of the stream
6. SNC (NSNC, 6) - bubble point ( $^{\circ}$ R)
7. SNC (NSNC, 7) - dew point ( $^{\circ}$ R)
8. SNC (NSNC, 8) - vapor ratio of stream
14. Stream Variables Test Vector Control Card
15. Stream Variables Test Vector - allowable fractional error  
for each element of the SN vector
16. Stream Control Variables Test Vector Control Card
17. Stream Control Variables Test Vector
18. Preferred Stream Vector Control Card - the number of  
preferred streams to be assumed
19. Preferred Stream Vector
  1. KPS (I) - stream number of the streams to be assumed  
in the event of trial and error solution;  
the first element will be assumed first  
(I = 1, ..., NOKPS)
20. Vector of Components which are not to be used for  
Component Balances - length of vector appears on run  
control card
21. Component Parameters Control Card - the number of com-  
ponents, length of a component vector, reference tempera-  
ture







22. Component Parameters Vector - contains physical property data

1. CMPROP (N, 1) - Latent Heat of Vaporization at the Normal Boiling Point (BTU/LB)
2. CMPROP (N, 2) - Normal Boiling Point ( $^{\circ}\text{R}$ )
3. CMPROP (N, 3) - Heat Capacity of Gas at one atm (BTU/LB  $^{\circ}\text{R}$ ) or first constant in heat capacity equation
4. CMPROP (N, 4) - Second constant times 100
5. CMPROP (N, 5) - Third constant times  $10^5$
6. CMPROP (N, 6) - Heat capacity of liquid (BTU/LB  $^{\circ}\text{R}$ )
7. CMPROP (N, 7) - Vapor pressure in psia at reference temperature
8. CMPROP (N, 8) - molecular weight
9. CMPROP (N, 9) - critical temperature ( $^{\circ}\text{R}$ )
10. CMPROP (N, 10) - critical pressure (psia)

The stream variables must appear in mass units, mass fraction or degrees Rankine.

Originally in the PACER logic a stream was either completely known or unknown, calculated or not calculated. Therefore, the stream was simply flagged as such. All feed streams to the process had to be totally known. With the extension to the PACER program unknown flow variables are flagged as such by a negative number in the appropriate element of the SN or SNC stream vector. Totally unknown stream



vectors require only that the stream number be specified. Elements three through NSLMAX can be left as blank on the input card and DREAD will insert negatives in these elements. However, in order that zeros be meaningful to DREAD partially specified stream vectors must have all elements completed including negative numbers for unknown elements.

If LCOMP, the length of the CMPROP matrix rows, is zero the CMPROP matrix will not be read.

## 2. Adapting the Flowsheet for PACER Use

### Method of Handling Reactors

A dummy feed and product stream must be supplied when considering reactors. The introduction of just a dummy product stream produces negative flow variables, which the program interprets as representing an unknown. These dummy streams need only be considered when using the balance routine for component balances. They will not appear in an overall mass balance since their total mass flows are equal. These streams are flagged as dummy streams by entering 10000. in the element of the stream vector normally allotted to temperature. Stream and flowsheet data for the unit shown in Figure IV might be encoded as shown in Table IV for usage with PACER. The reaction may be component one and two reacting to form component three. Thus, only component three would appear in stream 2, the dummy feed stream. Similarly only the reactants appear in the dummy product stream.



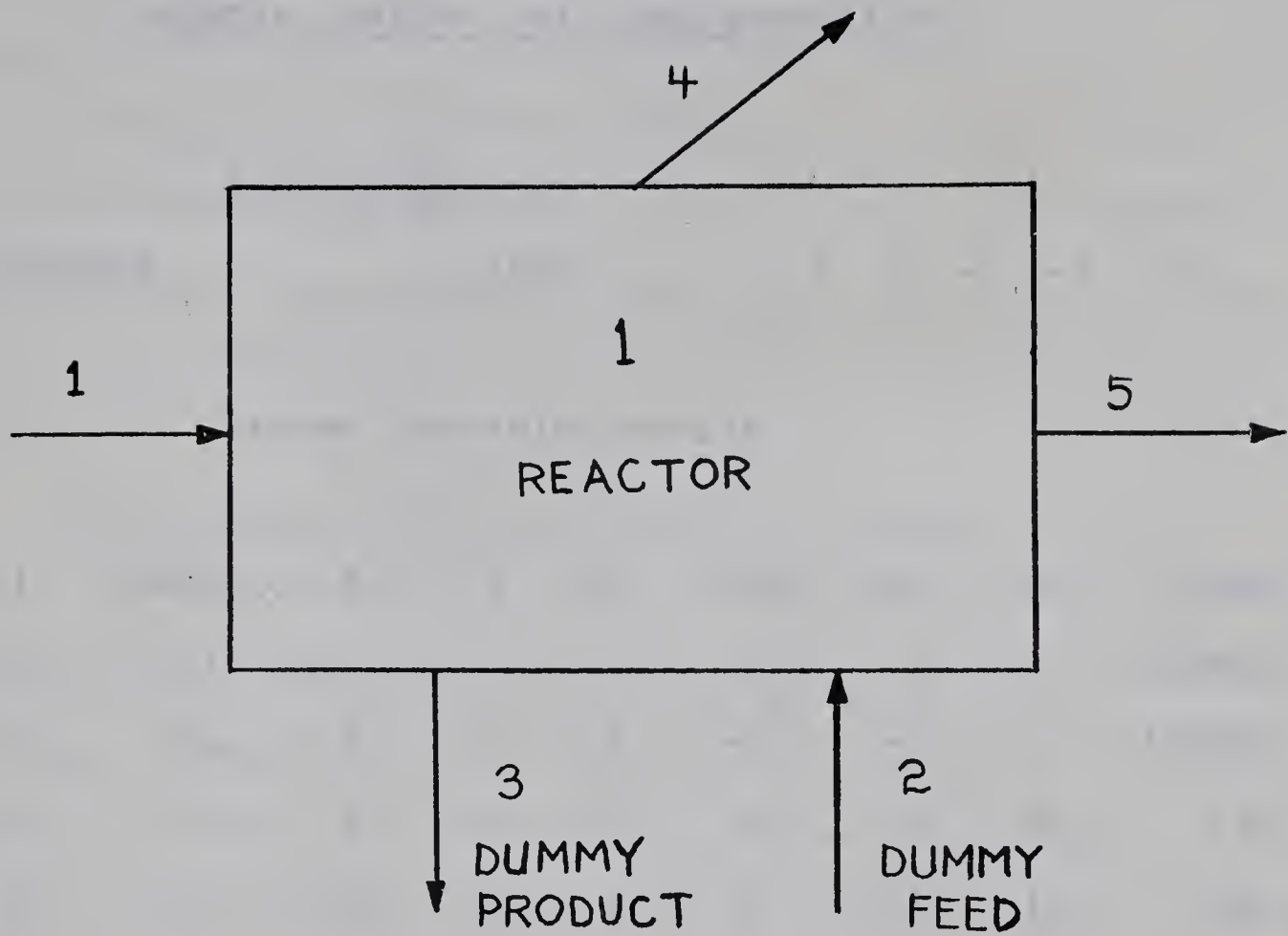


Figure VI  
Representation of a Reactor on a  
Process Flowsheet





TABLE IV

## Example Reactor Data Representation

## 1. Process Matrix

```
1UNAME6          -1002          1  2  -3  -4  -5
```

## 2. Stream Variables Matrix

|    |    |       |     |     |     |      |      |     |        |
|----|----|-------|-----|-----|-----|------|------|-----|--------|
| 1. | 1. | 1000. | .5  | .5  | 0.  | 500. | 500. | 0.  | 500.   |
| 2. | 1. | -1.   | 0.  | 0.  | 1.  | 0.   | 0.   | -1. | 10000. |
| 3. | 2. | -1.   | -1. | -1. | 0.  | -1.  | -1.  | 0.  | 10000. |
| 4. | 2. | 0.    | 0.  | 0.  | 0.  | 0.   | 0.   | 0.  | -1.    |
| 5. | 2. | -1.   | -1. | -1. | -1. | -1.  | -1.  | -1. | -1.    |



In order that the reactor subroutine UNAME6, be called only stream 1 must be totally known. Since NEXEQN (1) is negative the material and energy balance routine will not be called before the equipment subroutine, UNAME6. Stream 4 represents a heat flow and thus is zero mass. The element usually allotted for temperature in this vector will contain the heat generation within the reactor in BTU due to the reaction.

#### Method for Handling Distillation Column

If a unit such as a distillation column is involved in the process (see Figure V-a), streams 1, 7 and 4 would have to be known before the unit could be calculated by the PACER equipment subroutine if it were treated as three separate units. However, within certain specifications streams 4 and 7 need not be totally known. Therefore the unit is represented as shown in Figure V-b. The stream flows of 7 (if desired) and 4 can be calculated internal to the unit calculation and stored in appropriate vector elements. Stream 10 represents a stream of zero mass having a minus one (-1) in the last element if the heat input to the column is unknown. This represents the total heat flow from the column as a result of the reboiler, the overhead condenser and (if required) heat losses. Thus, although the unit can be involved in a material balance before it has been calculated, it cannot be used in an energy balance to solve for some unknown other than this unknown heat flow.



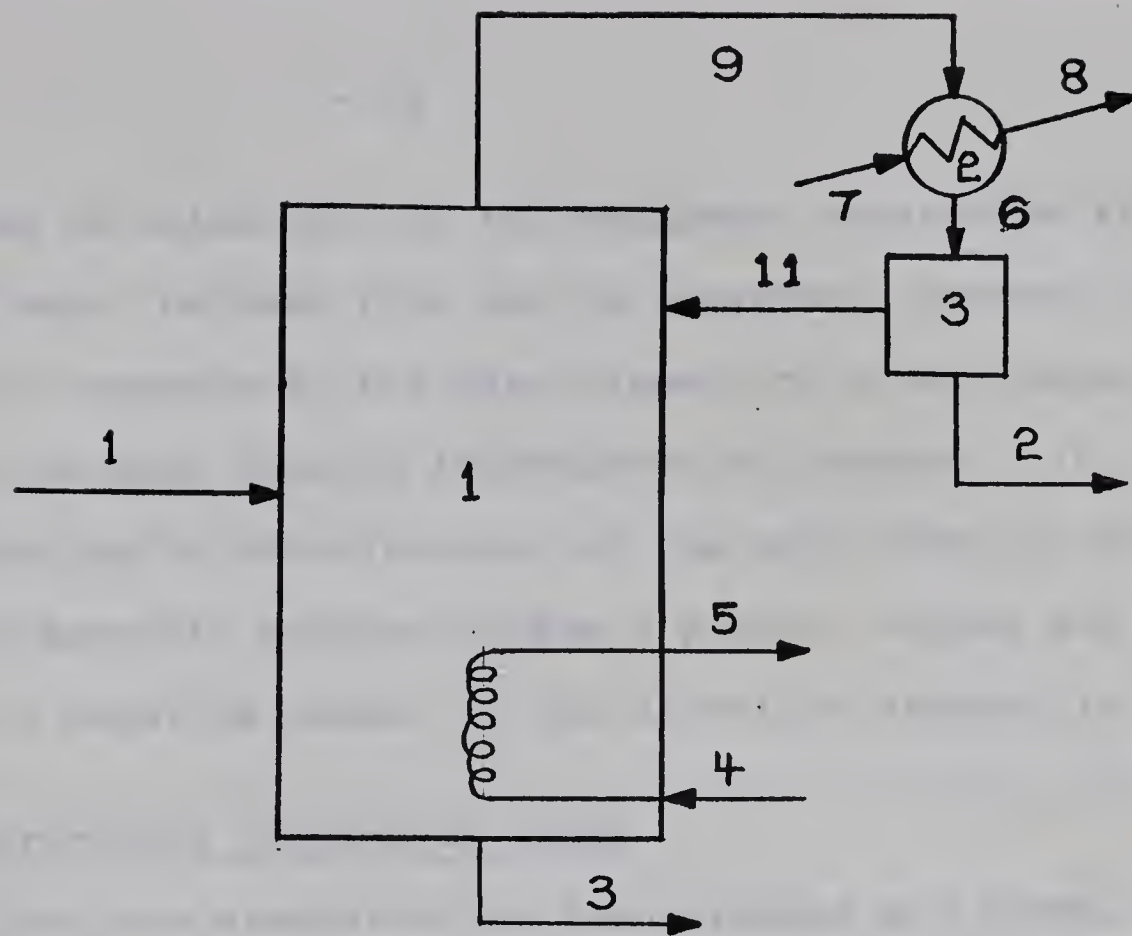


Figure V - a

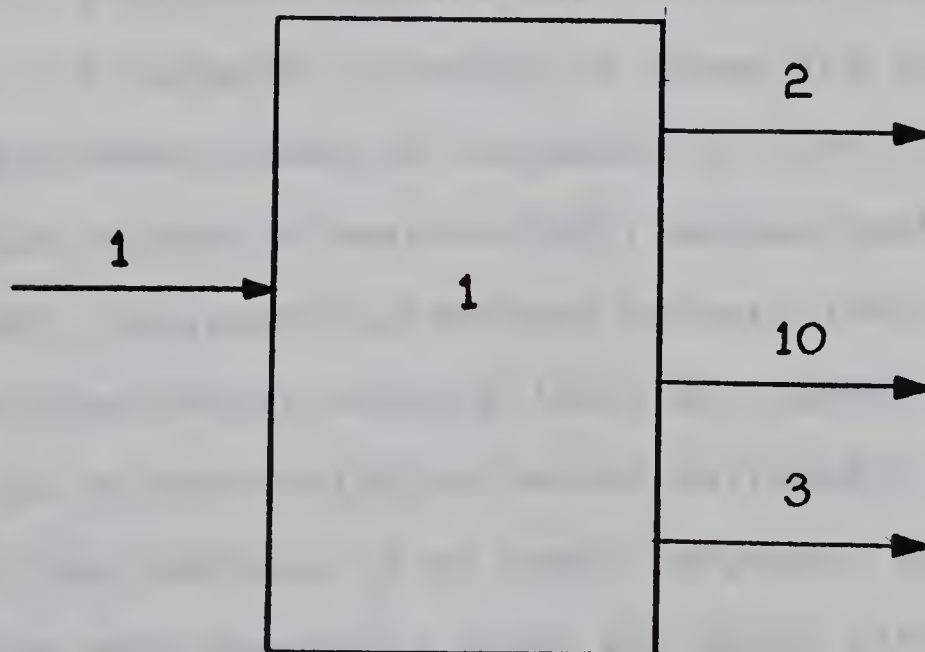


Figure V - b

Representation of a Distillation Column  
on a Process Flowsheet





The heat flow is calculated by the equipment subroutine if necessary. Note the heat flow can be negative. However, if minus one (-1) appears in the last element of a zero mass stream then the heat flow is interpreted as unknown. If there is doubt as to the direction of the heat flow, it is desirable to make the related stream a product stream and it will become a negative number if the direction assumed is wrong.

### 3. Restrictions Imposed by PACER

Since the dimensions of the matrices are fixed, the number of streams and equipments to be numbered is restricted. All stream numbers must be positive and less than sixty-one. Similarly all equipment numbers must be positive and less than thirty-one. If numbers in excess of these are to be used the dimension statements must be increased in size. The lengths of the stream variables vectors (SN), stream control variables vectors (SNC), equipment parameters vectors (EN), and equipment control parameters vectors (ENC) are fixed. Because the length of the stream variables vector is twenty, PACER is restricted to the analysis of an eight component system. However, storing both component flows and total flows with mass fractions is repetitious and one or the other can be discarded thus increasing the allowable number of components to sixteen without changing the dimensions of the matrices. Similarly the dimensions of the process matrix limit the number of streams associated with any one equipment to nine.



Virtually all of the available 32K core storage of the IBM 7040 was utilized with the integrated program. With the introduction of larger problems and more sophisticated equipment subroutines a chaining technique will have to be used. This is the method used at McMaster University during the simulation of a Sulphuric Acid Plant(23).

B. Integration of the Balance Routine into the PACER Logic

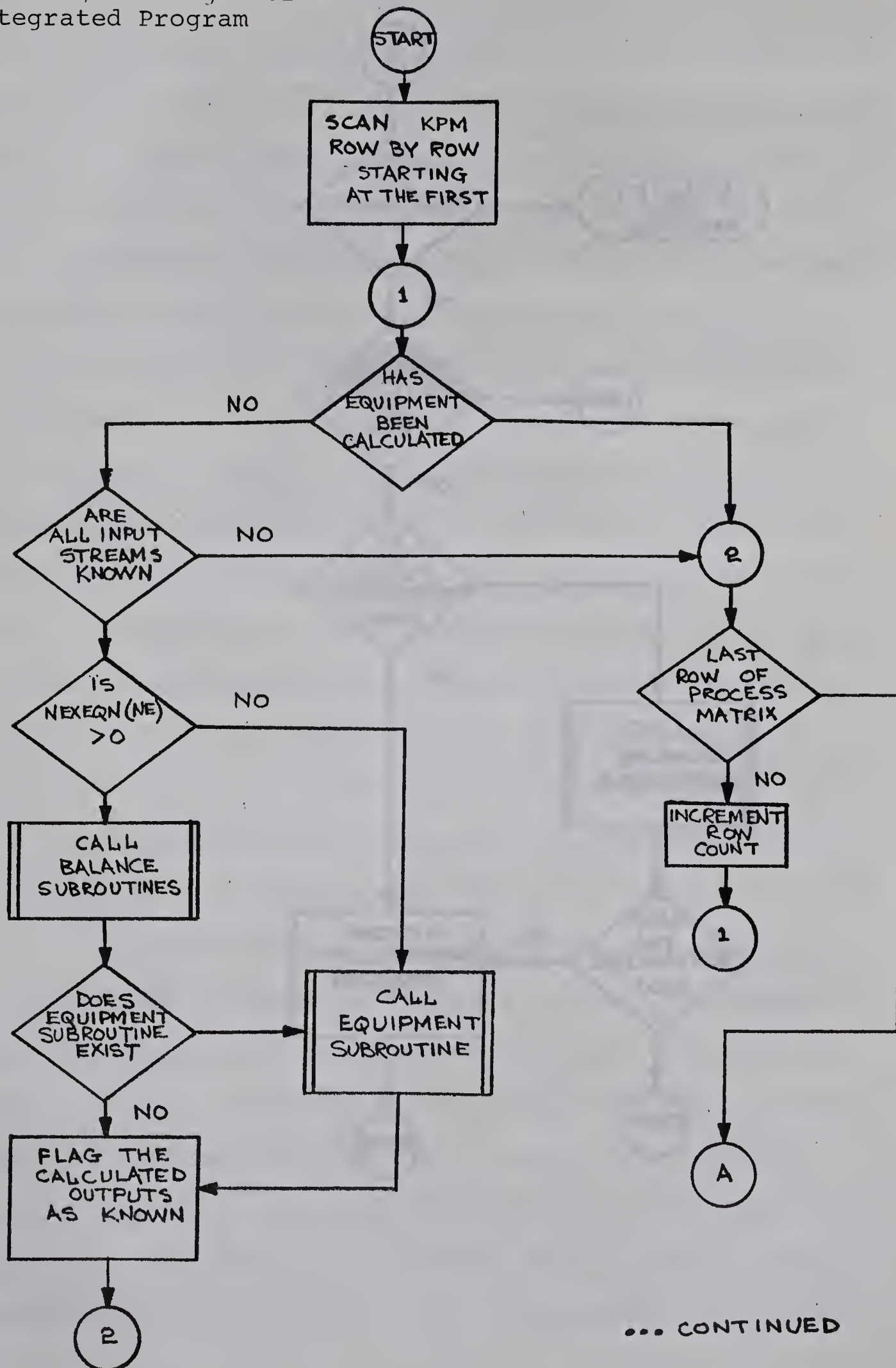
The process matrix discussed briefly in the section reviewing the literature and in the previous section encodes the flow diagram. Reiterating the basic rule of calculation, an equipment subroutine is used to calculate physical outputs from physical inputs for all units. The status of calculation of each stream and equipment is contained in the KSFLAG and KEFLAG vectors respectively. Initially both are set to zero. Any totally known streams are flagged greater than zero by the subroutine SCAN. Product streams are also flagged greater than zero since they are not involved in the calculation of the process with PACER.

The process matrix (KPM) is scanned to determine if any unit has all its input streams flagged known (see Figure VI). If such a unit is found the equipment subroutine is called and the unit calculation is executed. Upon completion of this step both this unit and its output streams are flagged known. Scanning of the process matrix is continued until all



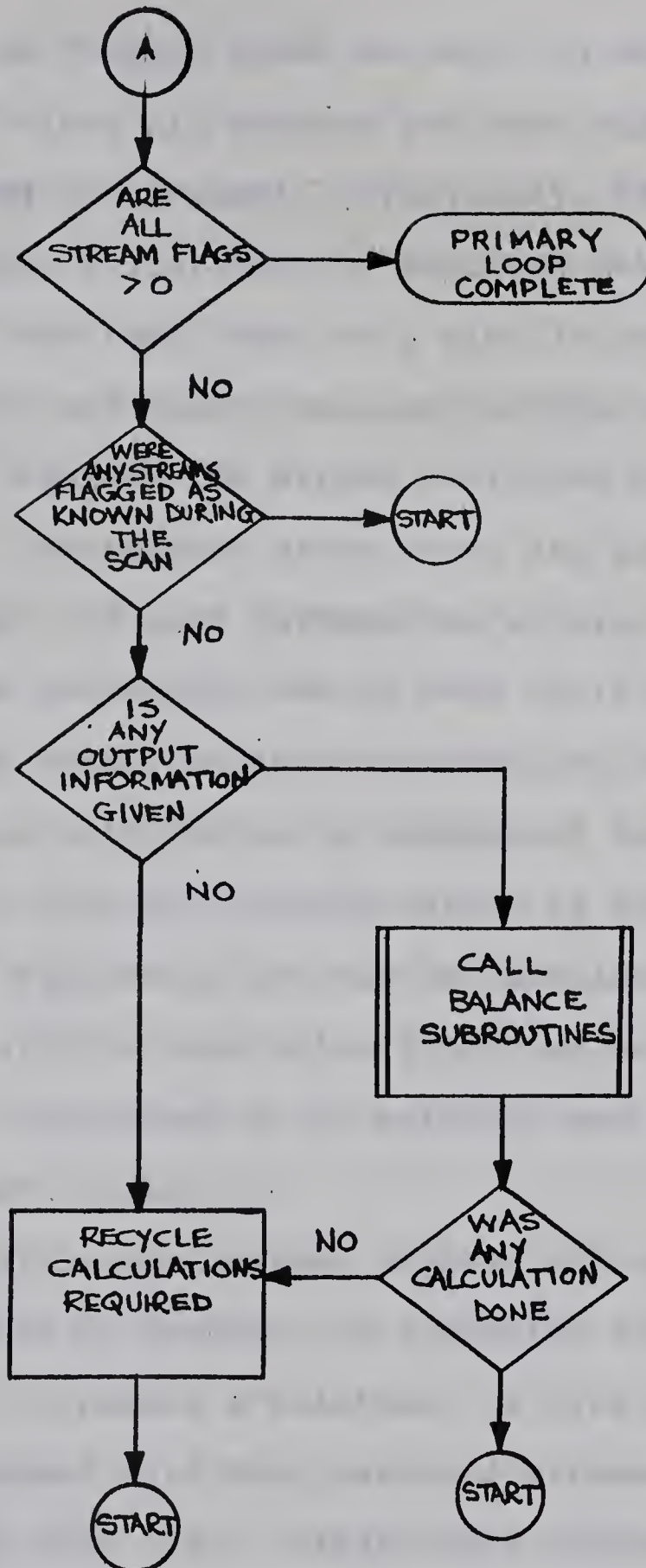


Figure VI - Flow Diagram  
Illustrating the Logic of  
the Integrated Program











streams and equipments are flagged known or until no more calculations can be made. Unless all streams had been calculated this implied recycle loops are present. Previously, PACER resorted to trial and error calculation by assuming unknown input streams first one, two then three at a time if necessary. However, with the material and energy balance routine introduced into the executive system, the stream variables matrix (SN) is now examined for information given about any product or inter-equipment stream. If such information exists, a suitable balance envelope containing one or more units will be found and material and energy balance calculations will be attempted. If the balance calculation is successful and some computation is completed, then the process matrix is scanned again to see if any more equipments can now be calculated. This cycle is repeated until no more calculation can be completed by the equipment subroutines or by material and energy balances around one or more units.

If there are still some unknown streams and equipments implying the presence of recycle, an iterative trial and error scheme is used to pursue a solution. A list of unknown streams (KUS) is formed with the preferred streams from KPS vector at the head of this list. First these streams are assumed known one at a time, then two at a time and finally three at a time. The problem cannot be solved by PACER if the order of recycle is in excess of third order. Then the





assumed known stream or streams are temporarily flagged greater than zero, the process matrix is examined to see if any more equipments can be calculated. If some units can be calculated and if the assumed inputs are to be calculated outputs of the equipments which can now be calculated, then trial and error calculations are commenced. If not, then a new stream or set of streams is taken from the unknown stream vector and the search repeated.

When a set of streams has been assumed, the resulting equipments which can now be calculated form the KE2 list. All units which are unnecessary to the recycle calculations such as branch units are removed from this list.

The value of each assumed stream, which is calculated as an output of a unit involved in the recycle loop, is used for the next iteration. This is repeated until each stream variable converges within some specified percentage error. This has been changed from the original PACER program which employed the absolute error criterion for convergence.

Referring to Figure VI, before an equipment subroutine is called, material and/or energy balances can be effected if this is requested in the input data as discussed in the previous section. Then, if an equipment subroutine exists, it can be called to complete the unit calculation. If all output streams from a particular unit are known and the material and energy balance routine is to be used to cal-





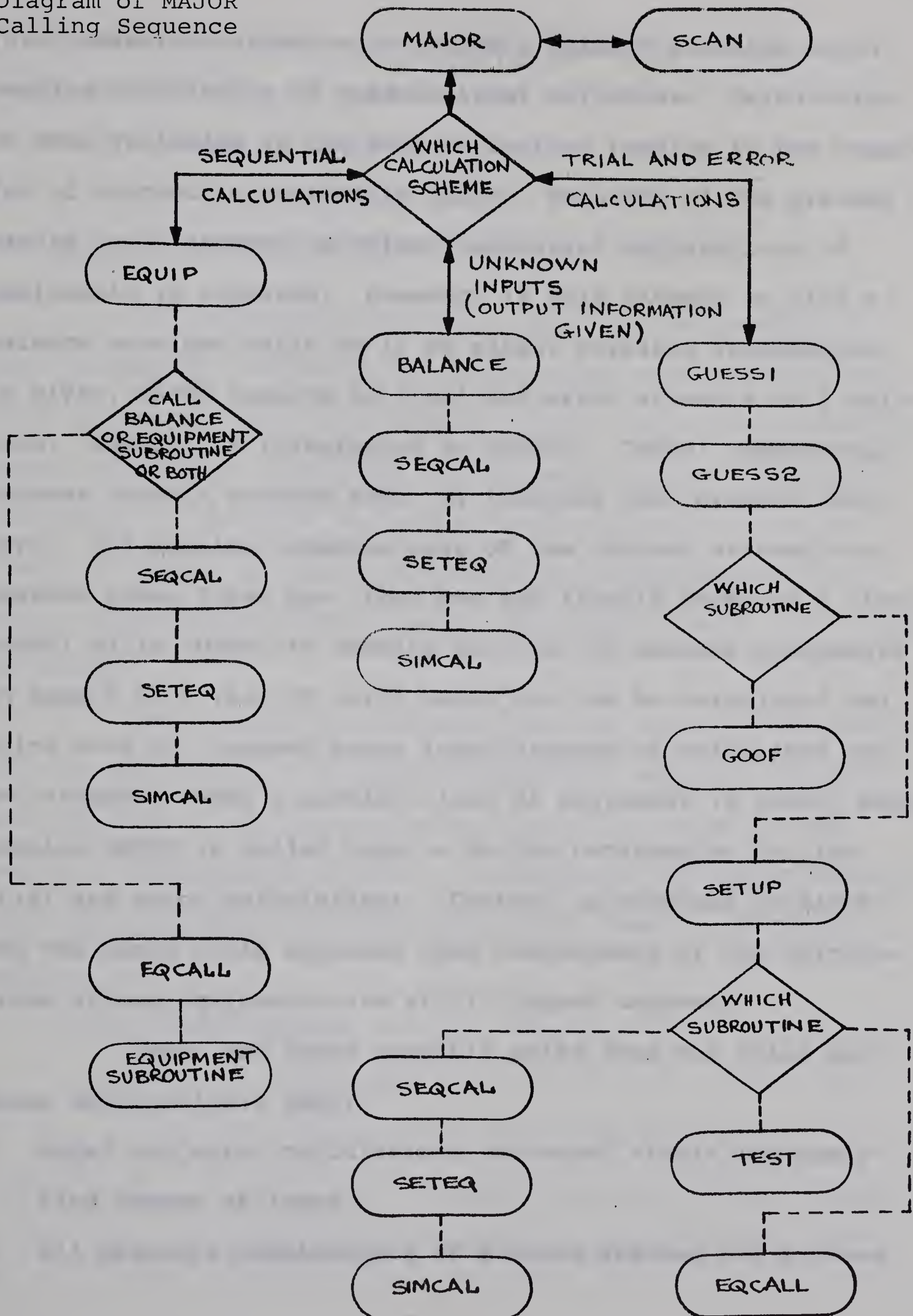
culate that unit, it is not called.

Reiterating the above discussion with reference to specific subroutines, the calling sequence is shown in Figure VII.

After the input phase of the program involving subroutines DREAD, DPRINT, and so on, the calculation phase is entered by calling subroutine MAJOR from the main calling routine, PACE. MAJOR zeros all stream and equipment flags then calls on subroutine SCAN to flag certain streams or equipments. For example, feed streams are flagged to plus seven (7) if totally known and product streams to plus ten (10). MAJOR now scans the process matrix attempting to find a unit with all input streams flagged known. If such an equipment is found, then subroutine EQUIP is called. Here it is determined whether the balance routine need be called, or the equipment subroutine or both. Since the unit around which the balance is to be performed is known the first subroutine of the balance routine, BALNCE, is bypassed. Upon completion of the unit calculation control is returned to MAJOR and the process matrix scan is continued. If, when no unit can be located with all its inputs known, there are still some equipments not flagged known, MAJOR examines output streams for some given information (that is inter-equipment streams or product streams flowing from units still flagged as unknown). MAJOR transfers control to subroutine BALNCE if some are located.



Figure VII - Flow Diagram of MAJOR Calling Sequence







This subroutine attempts to locate a balance envelope which permits calculation of unknown input variables. Calculation of some variables in the balance routine results in the transfer of control to subroutine MAJOR. Scanning of the process matrix in an attempt to effect sequential calculations of equipments is repeated. However, if this attempt to find a balance envelope fails or if no output variable information is given, MAJOR resorts to trial and error attempts at a solution. Control is transferred to GUESS1. GUESS1 temporarily assumes certain streams known by flagging them greater than zero, All possible combinations of the unknown streams are assumed known first one, then two and finally three at a time. GUESS1 calls GUESS2 to examine the list of unknown equipments in search of a list of units which can now be calculated and which have all assumed known input streams as calculated output streams. When a possible list of equipment is found, subroutine SETUP is called upon to do the bookkeeping for the trial and error calculations. Control is returned to MAJOR and the whole cycle repeated upon convergence of the calculations if any equipments are still flagged unknown.

There are three possible exits from the trial and error calculational loop:

1. trial and error calculations converged within the specified number of loops
2. all possible combinations of streams assumed and process





appears to be greater than third order recycle

3. trial and error calculations did not converge within the specified number of loops.

### C. Logic of Material and Energy Balance Routine

As indicated previously, this work deals with the use of material and energy balances around one or more units in an attempt to calculate unknown inputs using some specified output variables. The first problem which arises is that of location of a suitable balance envelope to compute unknown inputs. This is attempted by subroutine BALNCE (see Figure VII). Subroutine BALNCE will be discussed in more detail subsequently.

For any balance envelope and the streams associated with it the number of linearly independent material balance equations which can be written are equal in number to the number of components involved. In addition, an energy balance equation can supply an additional relationship. In general terms:

#### 1. Overall Mass Balance

$$\sum_{i=1}^N S_i * SN(i,3) = 0 \quad (14)$$

where

N = the number of streams involved in the balance

$S_i$  = sign function (always known)



= +1 if i is an input stream to the balance envelope, -1 otherwise

SN(i,3) = overall mass flow rate of stream i

## 2. Component Mass Balance

$$\sum_{i=1}^N S_i * SN(i,3) * SN(i,j+3) = 0 \quad (15)$$

or

$$\sum_{i=1}^N S_i * SN(i,j+3+NOCOMP) = 0$$

$$j = 1, 2, \dots, NOCOMP$$

where

SN(i,j+3) = mass fraction of component j

SN(i,j+3+NOCOMP) = component mass flow rate

## 3. Energy Balance

$$\sum_{i=1}^N \sum_{j=1}^{NOCOMP} S_i * SN(i,3) * SN(i,j+3) * H_j = 0 \quad (16)$$

or

$$\sum_{i=1}^N \sum_{j=1}^{NOCOMP} S_i * SN(i,J+NOCOMP+3) * H_j = 0$$

where

H<sub>j</sub> = enthalpy of pure component j at the temperature and pressure of stream i

Referring to equations 14, 15 and 16, the sign function (S<sub>i</sub>) is always known. However, any of the other variables, that is, overall mass flow rate, component mass flow rate, the mass fraction and the enthalpy may be unknown.



For the calculation of the enthalpy of a stream, mixing effects are neglected. The enthalpy of a solution at a given temperature, pressure and composition is given relative to the pure components at a reference temperature and at their saturation pressures. Where the normal state of aggregation of any component at this reference temperature and one atmosphere is gaseous, the reference pressure is taken as such(6).

$$H_i = \sum_{j=1}^{\text{NOCOMP}} X'_j H_j \quad (X'_j = \text{mass fraction}) \quad (17)$$

for  $p \geq$  bubble point pressure

$$H_j = \int_{T_{\text{ref}}}^T C_{p1j} dT - \int_0^{P_{0j}} \left( \frac{\partial H_j}{\partial P} \right)_{T_{\text{ref}}} dP + \int_0^P \left( \frac{\partial H_j}{\partial P} \right)_T dP \quad (18)$$

or for  $p \leq$  dew point pressure

$$H_j = \lambda_j - \int_0^{P_{0j}} \left( \frac{\partial H_j}{\partial P} \right)_{T_{\text{ref}}} dP + \int_{T_{\text{ref}}}^T C_{p_vj}^* dT + \int_0^P \left( \frac{\partial H_j}{\partial P} \right)_T dP \quad (19)$$

$\lambda_j$  = latent heat of vaporization of component j  
at reference temperature,  $T_{\text{ref}}$

$P_{0j}$  = saturation pressure of pure component j at  
reference temperature  $T_{\text{ref}}$

$p$  = pressure of stream i







$T$  = temperature of stream  $i$

$C_{p_v}^*$  = zero pressure heat capacity of vapor  $j$

$C_{p_l}$  = heat capacity of liquid  $j$

The pressure correction term in the above equations is evaluated using an equation of state. From thermodynamics:

$$dH = dU + v dP + P dv \quad (20)$$

$$dU = T dS - P dv \quad (21)$$

$$dH = T dS + v dP \quad (22)$$

From the Maxwell Relations:

$$\left(\frac{\partial S}{\partial P}\right)_T = - \left(\frac{\partial v}{\partial T}\right)_P \quad (23)$$

Therefore combining equations 22 and 23:

$$dH = v dP - T \left(\frac{\partial v}{\partial T}\right)_P dP \quad (24)$$

Using Van der Waal's equation of state for example purposes:

$$\left(P + \frac{a}{v^2}\right)(v - b) = RT \quad (25)$$

From the chain rule using  $T = f(P, v)$ :

$$\left[\left(\frac{\partial v}{\partial T}\right)_P dP\right]_T = - \left[\left(\frac{\partial P}{\partial T}\right)_v dv\right]_T \quad (26)$$

Substituting in (24) using (26):

$$(H^* - H)_T = RT - pv + \int_{\infty}^v \left[p - T \left(\frac{\partial P}{\partial T}\right)_v\right] dv \quad (27)$$



From equation (26):

$$\left(\frac{\partial P}{\partial T}\right)_v = \frac{\partial}{\partial T} \left( \frac{RT}{v-b} - \frac{a}{v^2} \right) \quad (28)$$

$$= \frac{R}{v-b} \quad (29)$$

Therefore:

$$(H^* - H)_T = RT - pv + \int_{\infty}^v \left( P - \frac{RT}{v-b} \right) dv \quad (30)$$

$$= RT - pv + \int_{\infty}^v - \frac{a}{v^2} dv \quad (31)$$

$$= RT - pv + \frac{a}{v} \quad (32)$$

However, in order to evaluate this pressure correction term, the specific volume must be evaluated. This can be done using the Newton-Raphson method for finding the root of equation 25.

$$F = \left( P + \frac{a}{v^2} \right) (v - b) - RT = 0 \quad (33)$$

$$F = Pv - Pb + \frac{a}{v} - \frac{ab}{v^2} - RT \quad (34)$$

$$F' = \frac{\partial F}{\partial v} = P - \frac{a}{v^2} - \frac{2ab}{v^3} \quad (35)$$

The constants a and b are calculated from the critical properties which part of the input data. The following iterative technique is used to solve for the specific volume.

$$v = v_o - \frac{F(v_o)}{F'(v_o)} \quad (36)$$



A more sophisticated equation of state might be desirable in some cases. However, a sophisticated equation of state with many constants can lead to large errors in derivatives. An equation like the Redlich-Kwong equation is perhaps the answer.

When a stream is in two-phase flow, the unit mass enthalpies of the liquid and vapor component phases are evaluated separately and added. Defining  $V$  as the vapor ratio or the pound moles of vapor per pound mole of mixture of stream  $i$  in Figure VIII:

$$V + L = F = 1 \quad (37)$$

$$z_j = y_j V + x_j L \quad (38)$$

$$\frac{y_j V}{y_j V + x_j L} = \begin{array}{l} \text{ratio of mass of component } j \text{ in vapor phase} \\ \text{to mass flow of component } j \text{ in stream } F. \end{array}$$

where

$x_j$  = mole fraction of component  $j$  in the liquid phase

$y_j$  = mole fraction of component  $j$  in the vapor phase

$z_j$  = mole fraction of component  $j$  in the stream  $i$

$$\frac{y_j V}{y_j V + x_j L} = \frac{\frac{y_j V}{x_j L}}{\frac{y_j V}{x_j L} + 1} \quad (39)$$





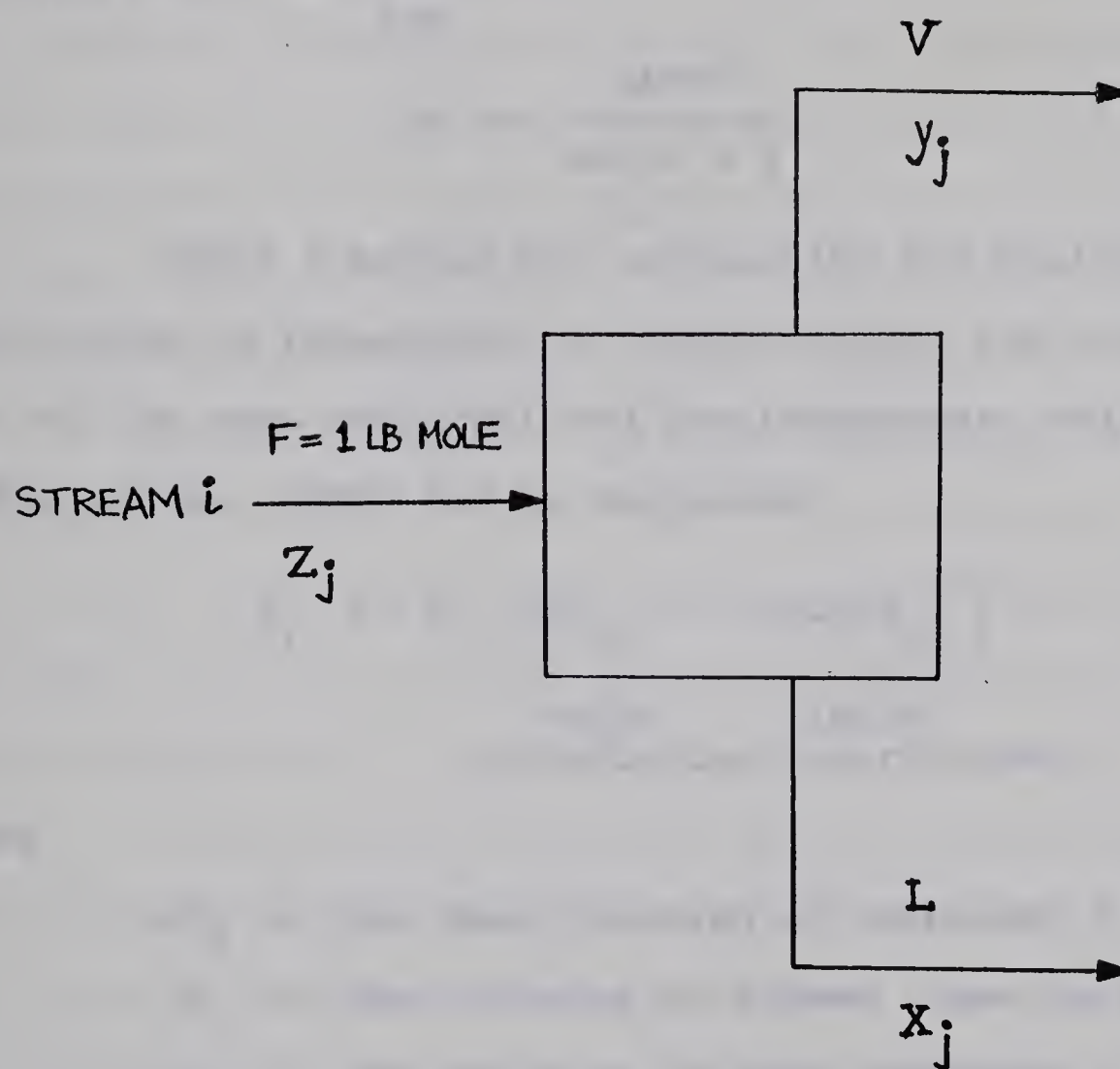


Figure VIII

Representation of Two Phase Flow



Since  $K_j = \frac{y_j}{x_j}$  :

$$\frac{y_j^V}{y_j^V + x_j^L} = \frac{\frac{K_j V}{1-V}}{\frac{K_j V}{1-V} + 1} \quad (40)$$

Defining  $RATIO = \frac{K_j V}{1-V}$

$$R = \frac{RATIO}{RATIO + 1} \quad (41)$$

Using a method for calculating the equilibrium ratio which is independent of concentration and only a function of the pure component and the temperature and pressure of the stream,  $RATIO$  can be evaluated.

$$H_i = z'_j \left[ \underset{\text{vapor contribution}}{R H_{Vj}} + \underset{\text{liquid contribution}}{(1-R) H_{Lj}} \right] \quad (42)$$

where

$z'_j$  = the mass fraction of component  $j$  in stream  $i$

$H_i$  = the enthalpy of stream  $i$  per unit mass

$H_{Vj}$  = the enthalpy of pure component  $j$  in the vapor phase per unit mass at the pressure and temperature of the stream  $i$

$H_{Lj}$  = the enthalpy of pure component  $j$  in the liquid phase per unit mass at the pressure and temperature of the stream  $i$





However, since there may not be an a priori knowledge of the quality and the quantity  $V$ , in most cases it will be desirable to divide a two-phase stream into two separate streams on the flowsheet. A pseudo-unit can be introduced to split the two-phase stream if desired. This will relieve the equipment subroutine of the unit into which this two-phase stream flows of calculating the compositions of the two phases, since the vapor and liquid compositions will be stored. Representing a two-phase stream by one stream on the flowsheet allows for storage of only the overall composition of the stream.

If the quality of a stream is specified then the quantity  $V$  must also be specified if the stream is in two phase flow. Subroutine QUAL calculates both the quality and the vapor ratio,  $V$ , using equilibrium ratios and the overall stream composition.

The dew point and bubble point temperatures of a stream are calculated after the composition and pressure of the stream is determined. The quality is calculated, if it is unknown, after the dew point and bubble point calculations have been completed. If the temperature of a stream is less than the bubble point temperature, then the quality is set to zero. If it is greater than the dew point temperature the quality is set to one. Otherwise a flash calculation is performed to determine the quality of the stream. This will be discussed further when subroutine QUAL is presented.



The quality, temperature and pressure of a stream must be known as well as its composition before the enthalpy of a stream can be determined by the subroutine available to the balance routine. An additional equation in solving for unknown mass flow variables can be supplied by the energy balance but all enthalpies and heat flows involved must be known so as not to introduce another unknown. Since the stream enthalpy is assumed to be a linear combination of pure component enthalpies at the temperature and pressure of the stream, the resulting energy balance equation is linear in the mass flow variables. The mass balance equations are likewise linear.

More often the case is that of solving the linear mass balance equations then solving for an unknown temperature of some stream involved in the balance. The energy balance equation is however non-linear in temperature and pressure. The unknown stream variable (temperature) is obtained by using the following iterative method:

$$x^{(n+1)} = x^{(n)} - \frac{f(x^{(n)}) (x^{(n)} - x^{(n-1)})}{f(x^{(n)}) - f(x^{(n-1)})} \quad (43)$$

where the problem is to find  $x$  such that  $f(x) = 0$ .

With reference to Figure IX suppose the unit enthalpies of streams 1, 3 and 4 are known with respect to a specified reference temperature.





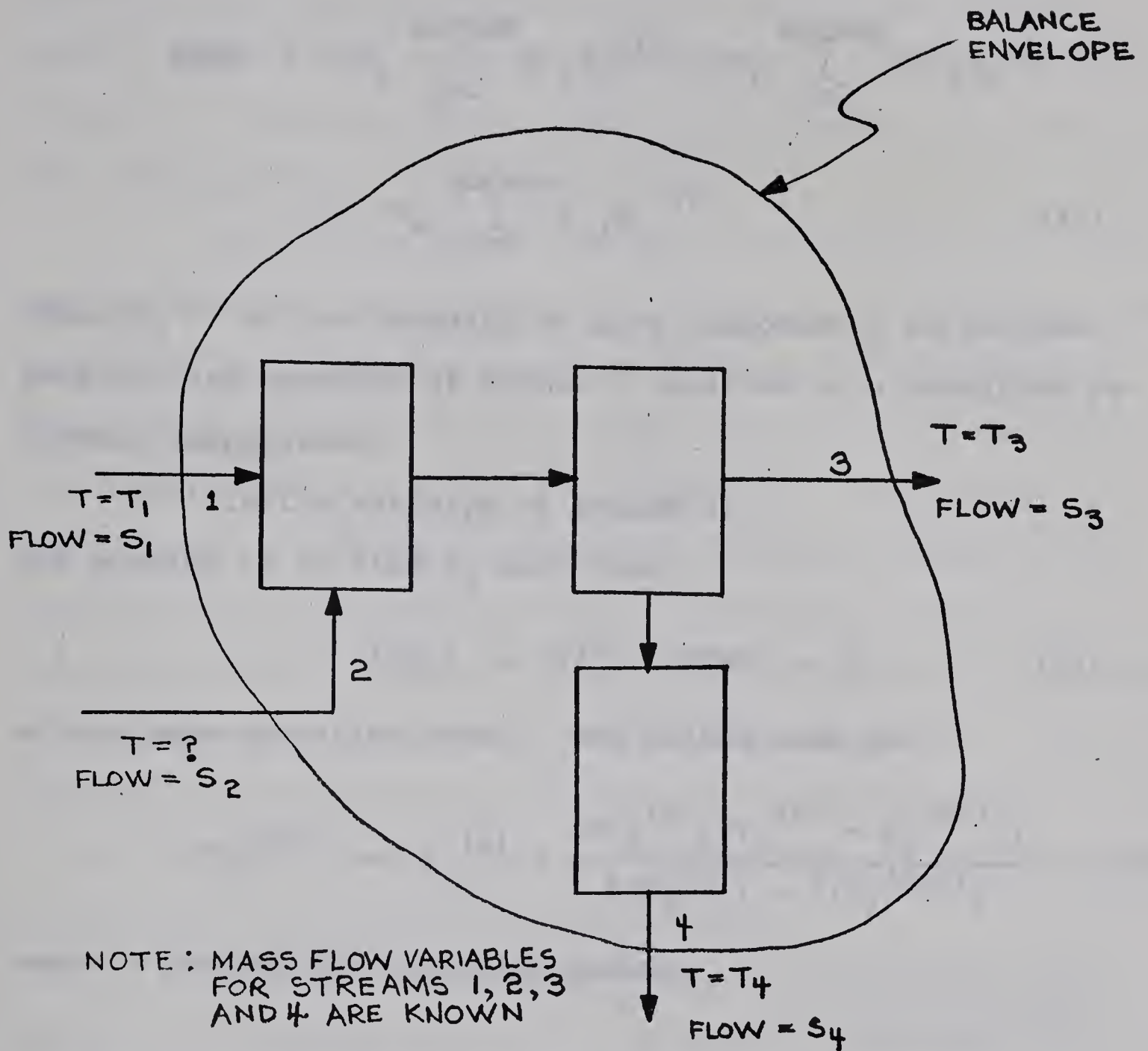


Figure IX

A Balance Envelope and Associated Streams





$$\begin{aligned} \text{HEAT} = & -S_1 \sum_{j=1}^{\text{NOCOMP}} x'_j H_j^{(1)} + S_3 \sum_{j=1}^{\text{NOCOMP}} x'_j H_j^{(3)} \\ & + S_4 \sum_{j=1}^{\text{NOCOMP}} x'_j H_j^{(4)} \end{aligned} \quad (44)$$

where  $H_j^{(i)}$  is the enthalpy of pure component  $j$  at the temperature and pressure of stream  $i$  relative to a specified reference temperature.

$H^{(2)}$  is the enthalpy of stream 2.

The problem is to find  $T_2$  such that:

$$f(T_2) = H^{(2)} - \text{HEAT} = 0 \quad (45)$$

within some specified error. The method used is:

$$T_2^{(n+1)} = T_2^{(n)} - \frac{f(T_2^{(n)}) (T_2^{(n)} - T_2^{(n-1)})}{f(T_2^{(n)}) - f(T_2^{(n-1)})} \quad (46)$$

where  $n$  denotes the iteration number.

#### D. The Material and Energy Balance Computer Program

The balance routine searches for unknown inputs if there is some output information given. Subroutine BALNCE executes this search. Having found a plausible group of units around which material and energy balances may be performed, subroutines SETEQ, SEQCAL, and SIMCAL attempt calculations. Enthalpies, equilibrium ratios, dew points, bubble points, and



qualities are calculated if desired by subroutines ENTHPY, EQUILK, DEWPT, BUBPT and QUAL respectively. The purpose and contents of these subroutines will be discussed in more detail subsequently.

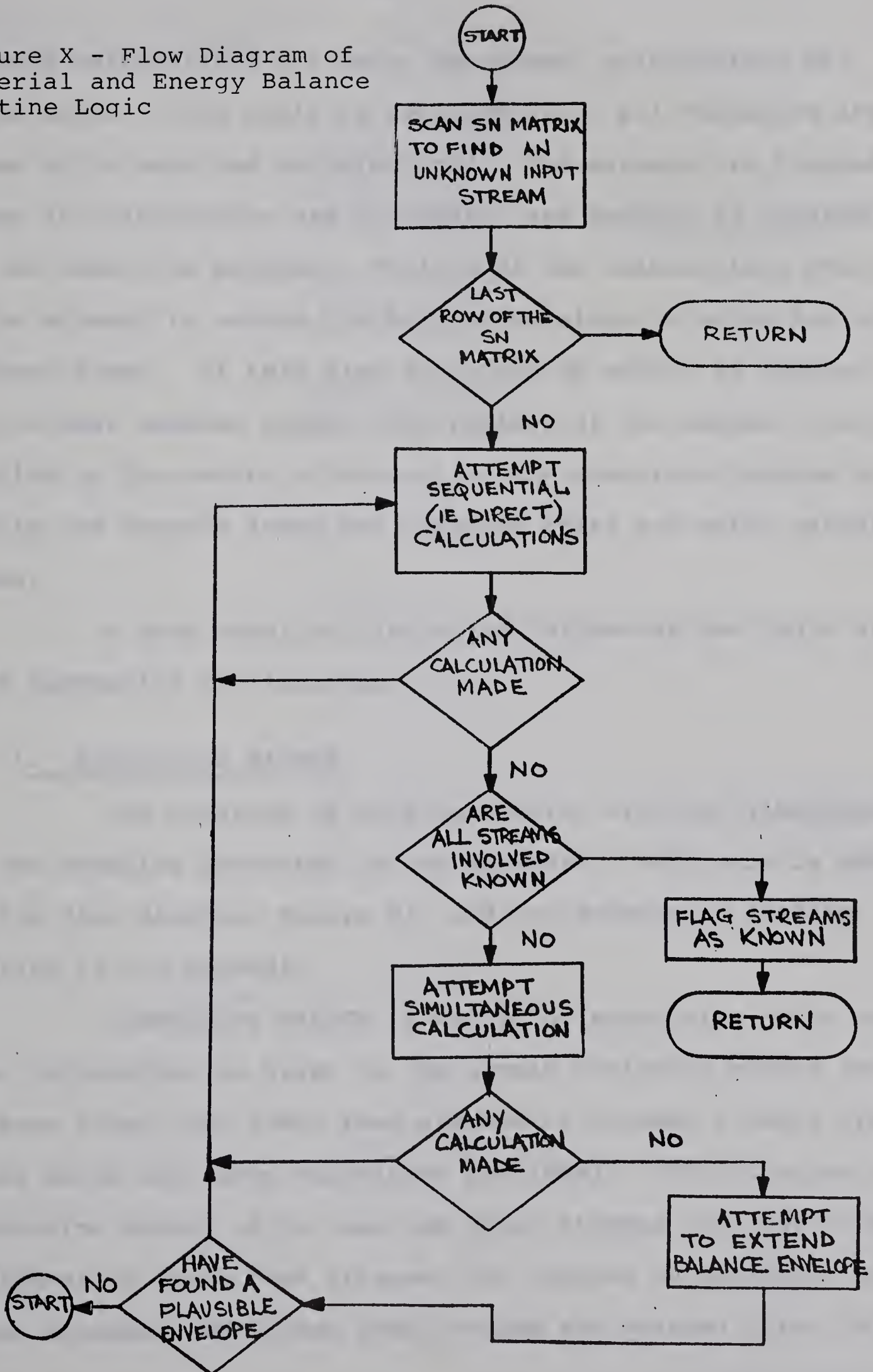
The balance subroutines as written handle only linear equations except when solving for an unknown temperature. The linearity is maintained for energy balances by assuming the enthalpy of a mixture to be a linear combination of its pure component enthalpies at the same temperature and pressure as the mixture. Similarly equilibrium ratios are assumed to be concentration independent when calculating the enthalpy of a two phase stream. That is the equilibrium ratios are functions of temperature and pressure alone for each component. The assumption of concentration independence of the equilibrium ratios is not necessary to the other calculations employing them such as dew point and bubble point calculations.

Referring to Figure X, after the sequential calculation of all possible units in the process (see Figure VI and VII), the balance routine is summoned to calculate unknown input streams. When an unknown input is located, a balance envelope is determined and an attempt is made to solve for unknown variables in the associated streams by direct calculations (that is one equation in one unknown). If this fails to render all variables known then an attempt is made to set up a set of simultaneous equations to yield some results. If some simul-





Figure X - Flow Diagram of  
Material and Energy Balance  
Routine Logic





taneous calculations are made, the direct calculations are tried again. This cycle is repeated until all variables are known or no more can be calculated. The streams are flagged known if calculations are successful and control is returned to the executive program. Failure of the calculations results in an attempt to extend the balance envelope to solve for this unknown input. If this also fails the SN matrix is scanned for another unknown input. The failure of the balance routine results in the return of control to the executive program to locate the recycle loops and commence trial and error calculations.

A more detailed discussion follows as the logic of each subroutine is discussed.

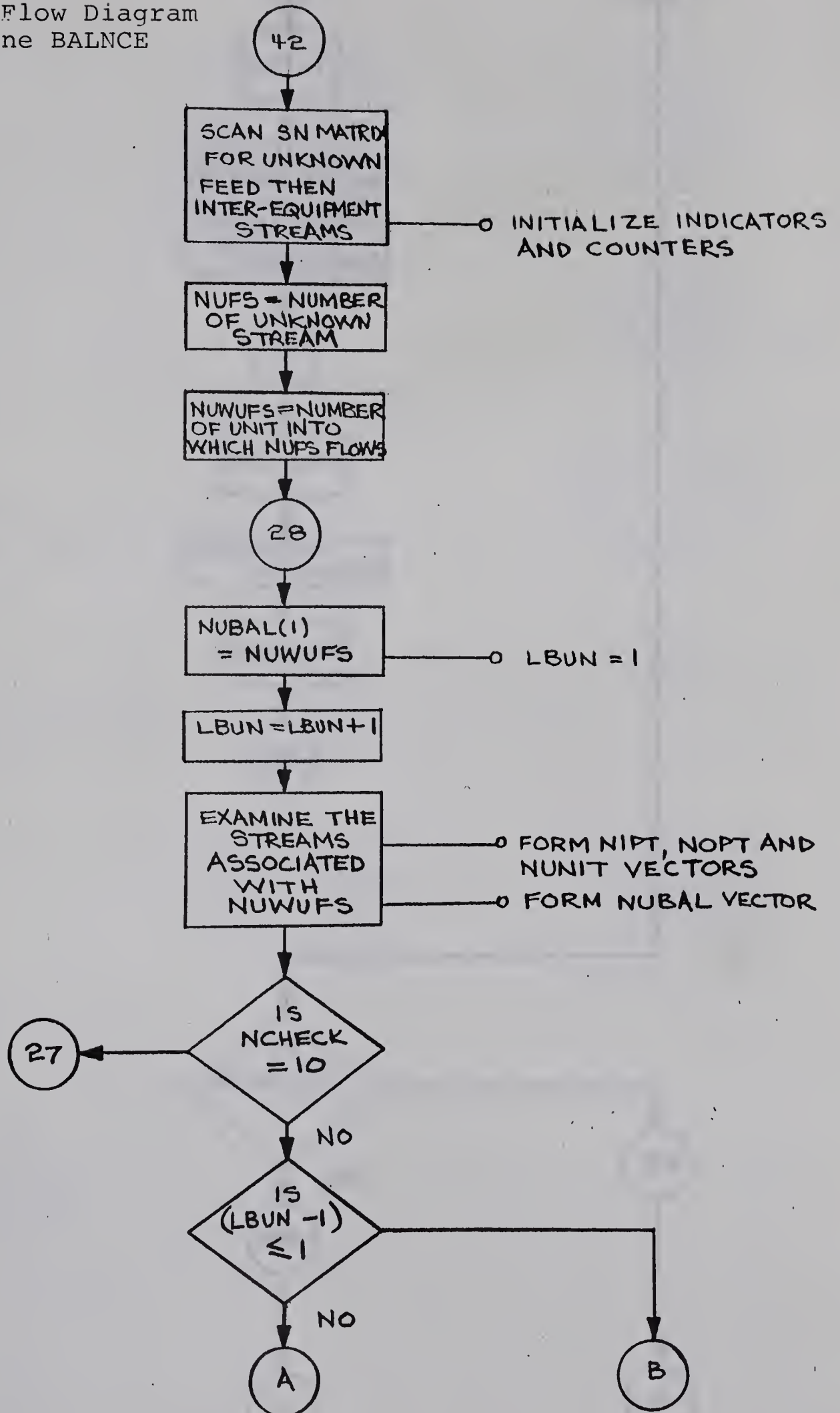
#### 1. Subroutine BALNCE

The workings of this subroutine will be illustrated in the examples presented in the Appendix. Reference is made to the flow diagram, Figure XI, and the subroutine FORTRAN listing in the Appendix.

Subroutine BALNCE is called by subroutine MAJOR when some information is given in the stream variables matrix for streams other than fresh feed streams or streams flowing from units which have been calculated previously. The function of subroutine BALNCE is to scan the input streams (be they inter-equipment or fresh feed streams) for unknown or partially unknown streams. The fresh feed streams are scanned first followed



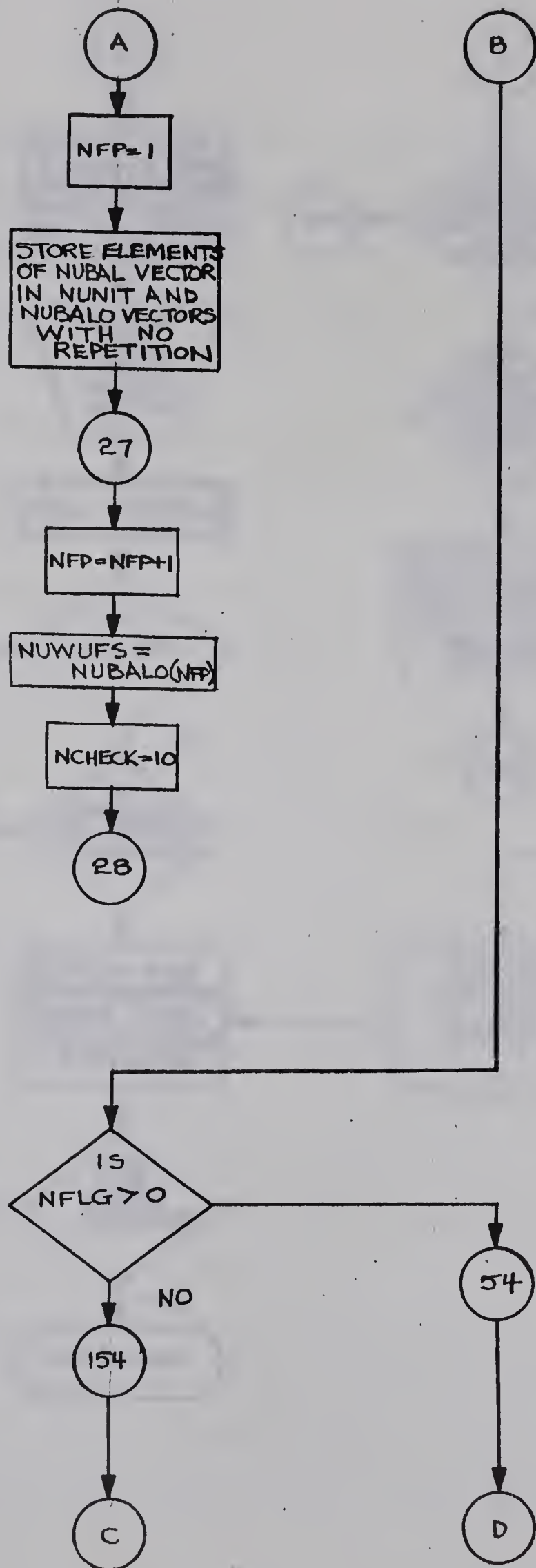
Figure XI - Flow Diagram  
for Subroutine BALNCE



... CONTINUED

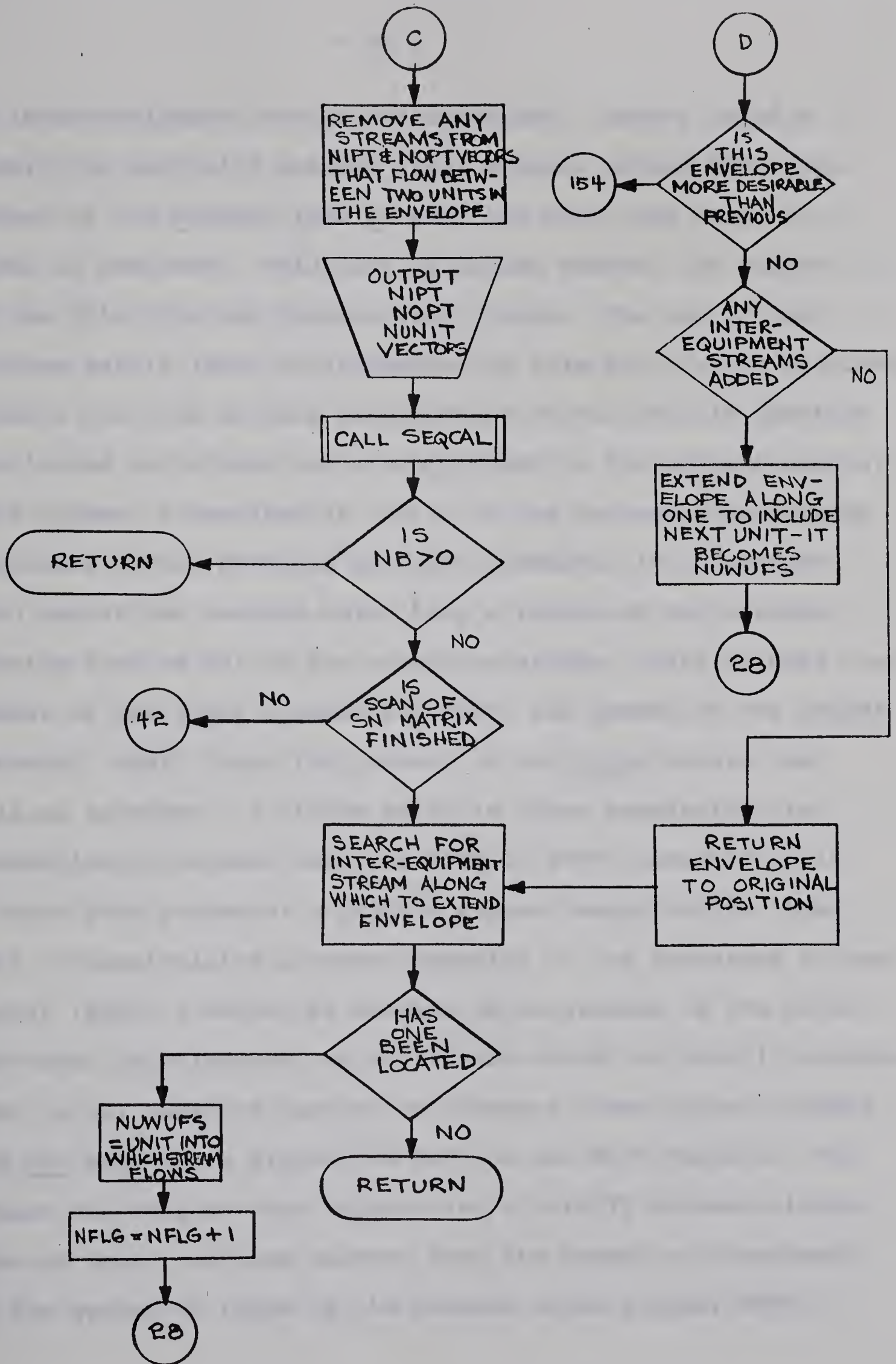






... CONTINUED









by inter-equipment streams, if necessary. Having found a totally or partially unknown input stream, called NUFS, the number of the unknown feed stream, the unit into which it flows is examined. This unit is called NUWUFS, the number of the unit with the unknown feed stream. The row of the process matrix (KPM) corresponding to this unit is now examined (recall that the streams associated with the unit in question are listed in columns two through N3MAX in the process matrix). Each stream is examined in turn. If the corresponding stream variables vector contains positive elements, it is entered into one of the vectors which keep a record of the streams flowing into or out of the balance envelope. NIPT harbors the number of the input streams and NOPT, the number of the output streams. NUNIT lists the numbers of the units within the balance envelope. A stream which is under examination is automatically entered into the NIPT or NOPT vector if it is a fresh feed stream or a product stream respectively. Numbers of uncalculated streams appearing in the preferred stream vector (KPS), a vector of streams to be assumed in the trial and error calculations, or of streams which are totally unknown, that is all negative numbers in elements three through NSLMAX, are not entered in either the NOPT or the NIPT vectors. The reason for this is that introducing a totally unknown stream adds at least one more unknown than the number of components in the system to those of the unknown input stream, NUFS.



Thus, enough equations cannot be written to solve for the unknowns involved. The numbers of the units into which, or from which, these rejected streams flow are entered into the NUBAL vector to be considered subsequently when all the streams associated with NUWUFS have been considered. Then elements of NUBAL are transferred to NUBALO. Each element in NUBALO is then set to NUWUFS and its associated streams examined in the manner analagous to that described above. The cycle is repeated until no new stream numbers have been added to NUBAL and all the elements of NUBALO have been considered. This will occur when all the associated streams crossing the boundary of the balance envelope are fresh feed streams, product streams or totally or partially known inter-equipment streams. Of course, NUFS may or may not be a totally unknown stream.

Subroutine SEQCAL is now called in an attempt to effect the balance calculation. If the balance calculation fails, control returns to BALNCE. The flag, NB, is zero upon return when the calculation has failed.

Subroutine BALNCE now attempts to extend the balance envelope along inter-equipment streams flowing across the boundary of the envelope. Choosing the first inter-equipment output stream from NOPT which is not totally known, if any, the unit into which it flows now becomes NUWUFS and the analysis continues. If extension along this path leads to a more desirable balance envelope, that is, the number of unknowns





decreases or the number of components involved increases, this new envelope is used and it in turn is extended in the identical manner if the calculations fail again. The boundary is returned to its original position if the extension is not desirable and search for another inter-equipment stream along which to extend the envelope is made.

The routine may not always find a balance envelope for which calculation is successful. In this event control is returned to MAJOR and trial and error calculations are attempted. If successful, however, control is returned to MAJOR where the process matrix is rescanned in search of a unit with all input streams now known. Refer to Figures VI and VII.

## 2. Subroutine SEQCAL

Accompanying the following discussion is the flow diagram, Figure XII, which indicates the function of SEQCAL. Again the FORTRAN listing is given in the Appendix.

The stream numbers stored in the NOPT and NIPT vectors are transferred to the NSTR vector. This vector contains all the numbers of the streams associated with the balance. The stream variables vectors for these streams are examined for unknown variables (negatives in appropriate elements) and attempts are made to solve for these directly. That is, SEQCAL searches for one equation involving one unknown. This is done by examining appropriate elements of the SN matrix and applying the following equations:





1. Overall Mass Balance (N = number of streams crossing the boundary of balance envelope)

$$\sum_{i=1}^N S_i * SN(i,3) = 0 \quad (47)$$

where

$$S_i = \pm 1$$

2. Component Mass Balances

$$\sum_{i=1}^N S_i * SN(i, j+NOCOMP+3) = 0 \quad (48)$$

$$j = 1, 2, \dots, NOCOMP$$

where

$SN(i, j+NOCOMP+3)$  is the  $j^{th}$  component mass flow rate in stream  $i$

3. Summation of Component Flows

$$\sum_{j=1}^{NOCOMP} SN(i, j+NOCOMP+3) = SN(i, 3) \quad (49)$$

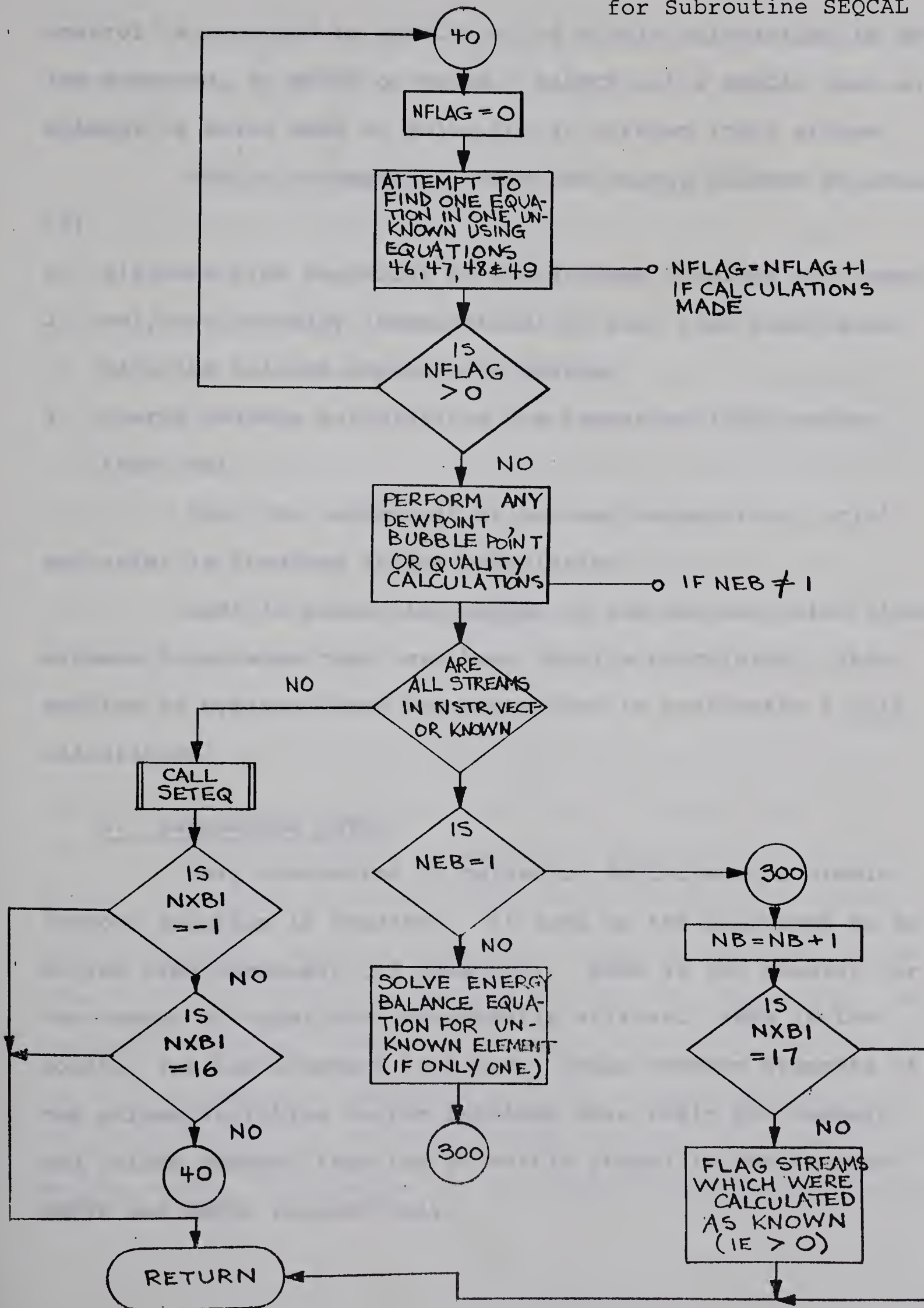
4. Mass Fraction Relationship

$$SN(i, j+3) = \frac{SN(i, j+NOCOMP+3)}{SN(i, 3)} \quad (50)$$

If any calculations are made on the first pass, the search is repeated. Subroutine SETEQ is called if all the variables are not solved for in SEQCAL. Control is returned to SEQCAL from SETEQ. The search for variables which can be solved for directly is repeated if the simultaneous calculations were successful. When calculations fail in both SEQCAL and SETEQ,



Figure XII - Flow Diagram for Subroutine SEQCAL







control is returned to BALNCE or, if a unit calculation is being executed, to SETUP or EQUIP. BALNCE calls SEQCAL when an attempt is being made to solve for an unknown input stream.

SEQCAL attempts to solve the energy balance equation if:

1. all mass flow variables of the streams involved are known
2. only one enthalpy (temperature) or heat flow associated with the balance envelope is unknown
3. energy balance calculations are requested (NEB greater than one).

When the unknown is an unknown temperature, trial and error is involved in the calculation.

Last in subroutine SEQCAL is the section which flags streams known when they have been totally calculated. This section is bypassed when the subroutine is performing a unit calculation.

### 3. Subroutine SETEQ

This subroutine is called by SEQCAL when a simultaneous solution is required. It sets up the equations to be solved simultaneously (if possible). NROW is the counter for the number of equations successfully written. KSTR is the counter for the unknowns involved. These unknown elements of the stream variables vector involved have their row numbers and column numbers from the SN matrix stored in the vectors NKSTR and NKCOL respectively.



Reference is made to the flow diagram of subroutine SETEQ, Figure XIII. The subroutine listing is found in the Appendix.

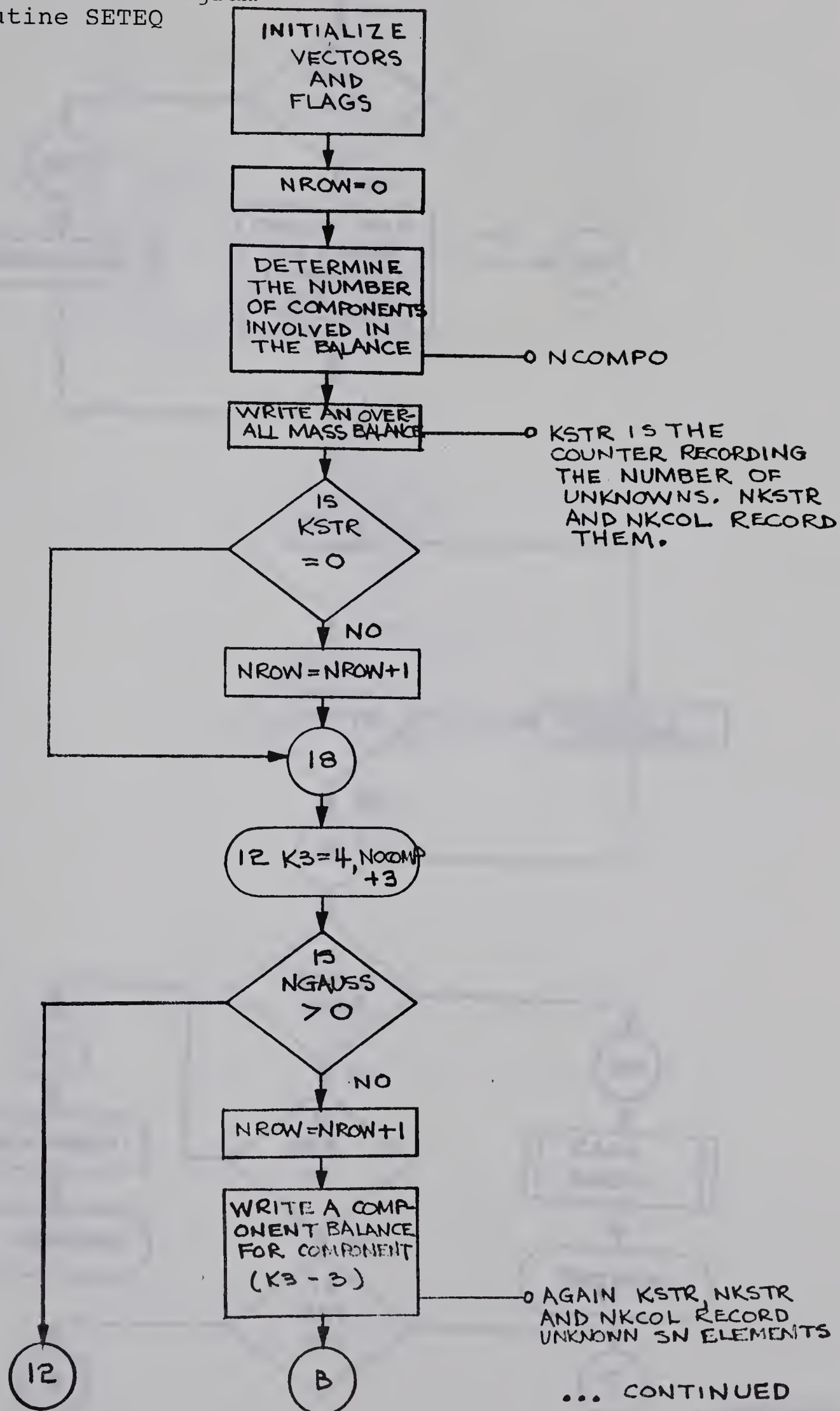
The first attempt to obtain an equation is the overall mass balance. Then component balances equal in number to one less than the number of components involved in the balance, NCOMPO, are used in attempts to obtain additional equations. If the number of unknowns involved, KSTR, equals the number of equations obtained, NROW, then subroutine SIMCAL is called to effect the solution of the simultaneous linear algebraic equations. In the event that KSTR exceeds NROW by one, an energy balance may be used to obtain the required equation. This is possible if the temperature, pressure and quality of all the streams involved is known so as not to introduce another unknown with the new equation.

If the component mass flow rate is known for each component in each stream or if either the overall mass flow rate or the mass fraction is known for each component in each stream, an energy balance can be attempted directly. However, NCOMPO component balances must be written and component flows used as unknown elements when the overall mass flow rate, some component flows and their corresponding mass fractions are unknown in any stream associated with the balance calculation. The energy balance can now be written subject to the above conditions again. The reason for this change in procedure is that



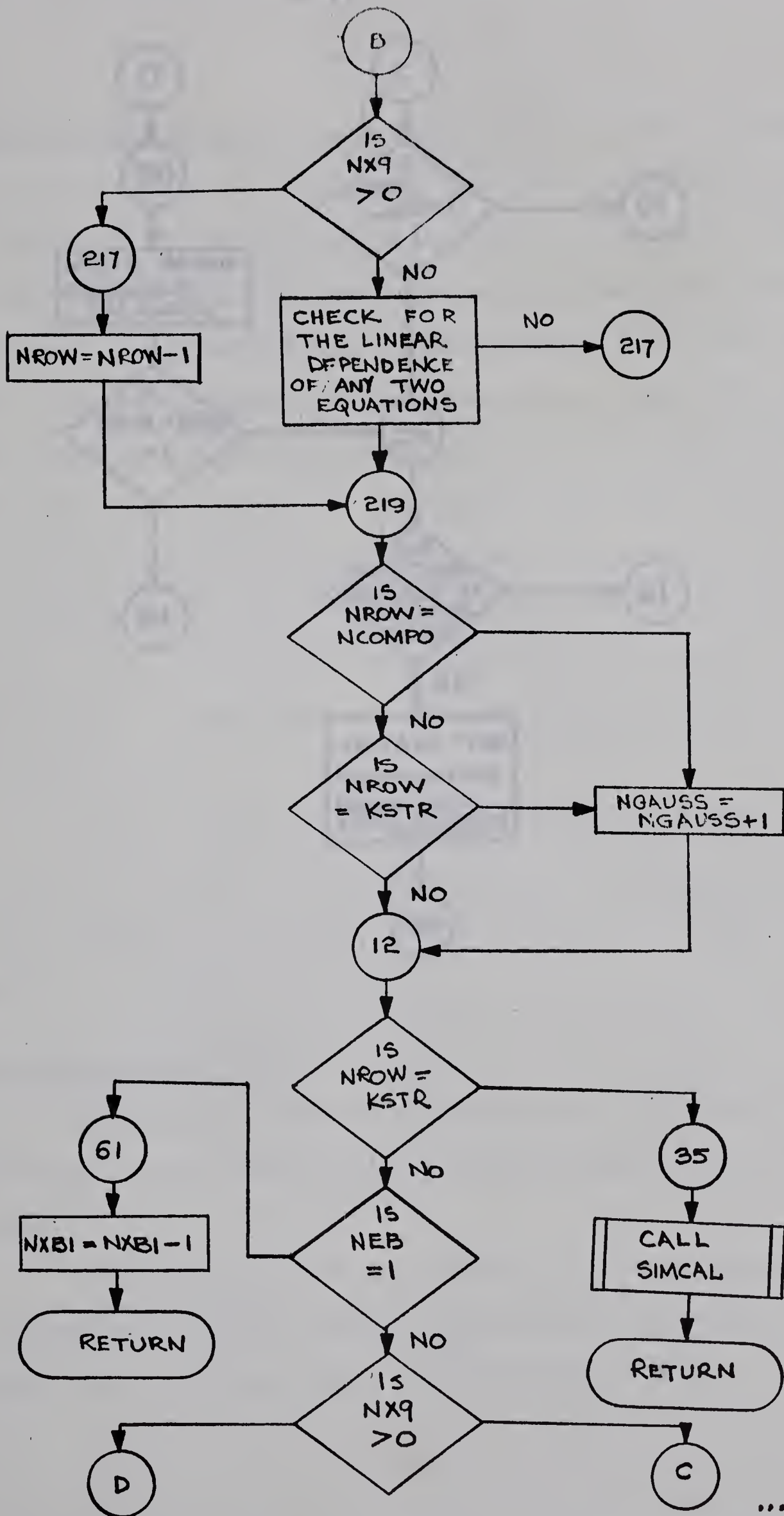


Figure XIII - Flow Diagram  
for Subroutine SETEQ

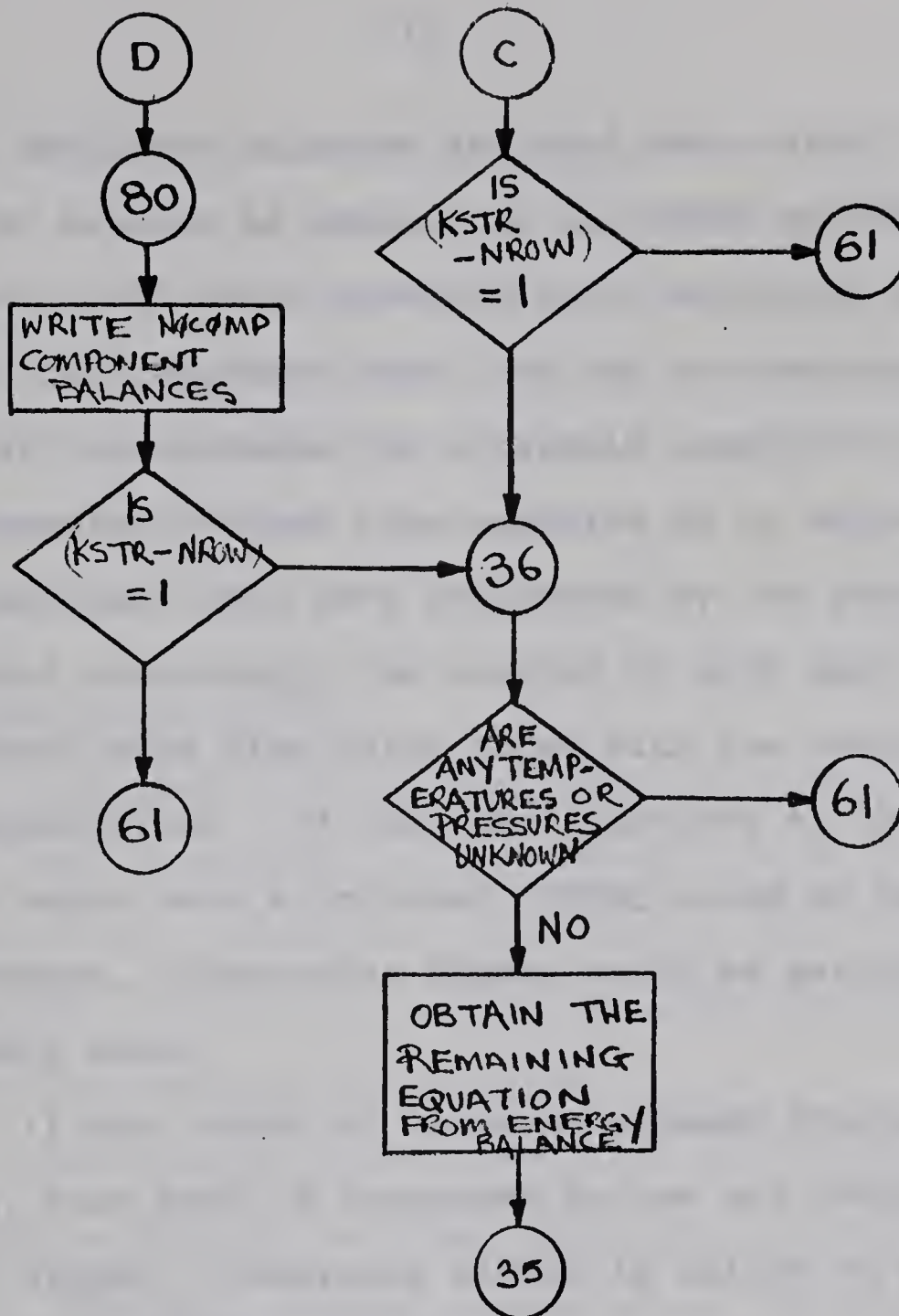
















if NCOMPO component balances are used the overall flow rate need not be entered as unknown in the NKSTR and NKCOL vectors. Thus, there would be no possibility of obtaining the overall mass flow, the component mass flow and the corresponding mass fraction all as unknowns for a certain component necessitating another equation derived from equation 49 to solve for each unknown component flow rate introduced by the energy balance. As mentioned previously, the storing of both mass fractions and component mass flow rates along with the overall mass flow rate is repetitious. If the mass fractions or the component mass flow rates were eliminated, SETEQ could be easily adapted to this change. Subroutine SEQCAL would be partially eliminated in this case.

If the number of unknowns exceeds the number of equations, then NXB1 is decreased by one and control is returned to SEQCAL. Otherwise SIMCAL is called to effect the solution of the NROW equations.

#### 4. Subroutine SIMCAL

Subroutine SIMCAL employs Gaussian Elimination with interchange to solve NROW linear algebraic equations in NROW unknowns.

Reference is made to any text on numerical analysis for a discussion of the method. To switch to another method requires only the replacement of subroutine SIMCAL.



If the equations cannot be solved, that is, if the interchange fails to produce an acceptable set of equations, control is transferred to SETEQ after NXB1 is diminished by one. An alternate routine for solving NROW linear algebraic equations should be supplied to handle such a case. The subroutine is provided with an error message to signal the existence of this problem.

#### 5. Subroutine ENTHPY

This subroutine calculates the enthalpy per unit mass of a pure component at a specified pressure and temperature relative to a reference temperature, TREF. Any other subroutine for calculating enthalpies could be used as long as it uses the above specifications.

The calculation of enthalpies was discussed previously. If NEB is flagged to three, pressure correction using Van der Waal's equation of state is neglected. Refer to Figure XIV.

The latent heats of vaporization given in the physical properties vectors (CMPROP) are given at the normal boiling point of the component in question. The latent heat of vaporization at the reference temperature is calculated from the Watson Correlation(5):

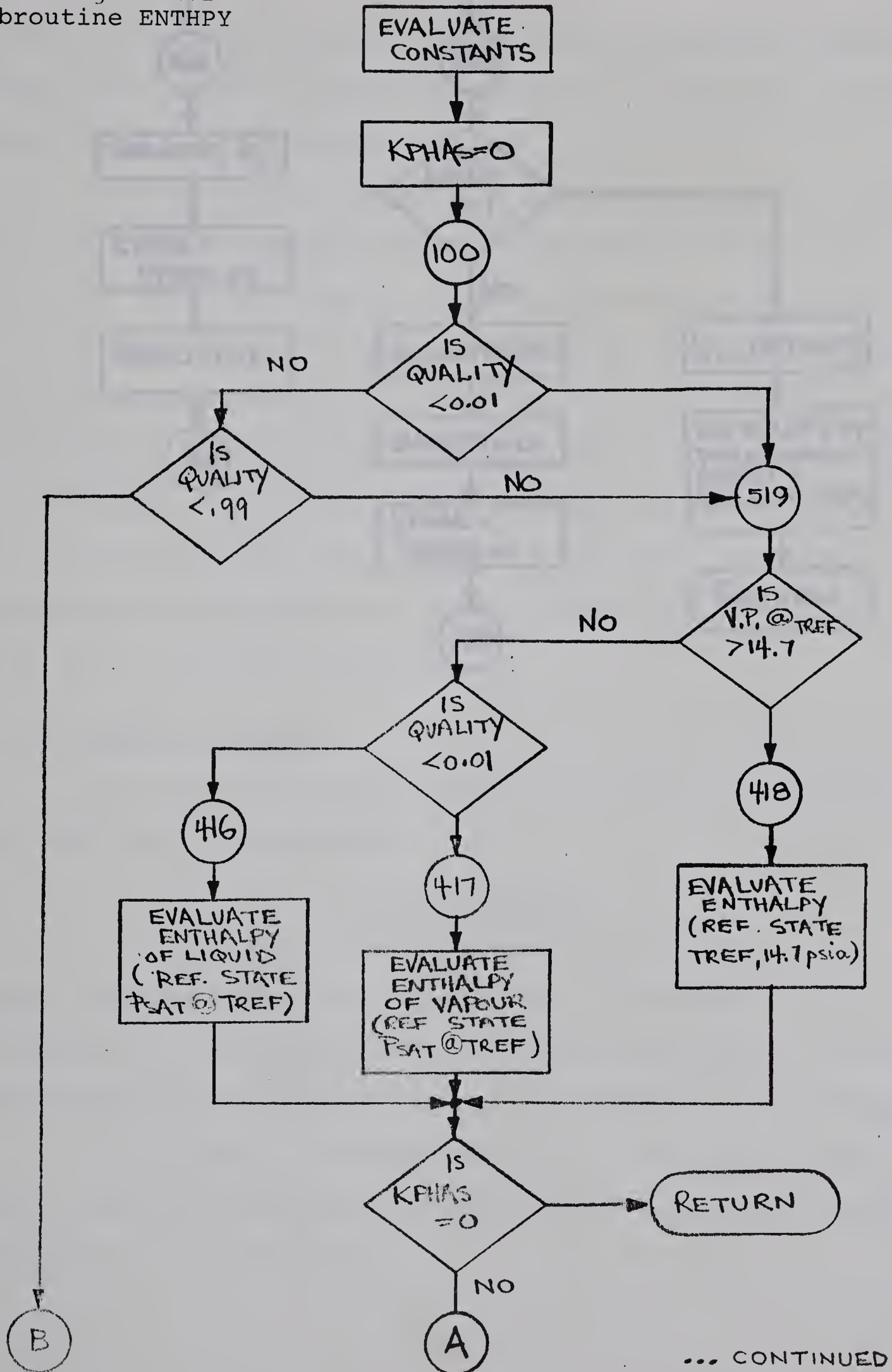
$$\frac{\lambda_{TREF}}{\lambda_{NBP}} = \left( \frac{1 - \frac{TREF}{TC}}{1 - \frac{NBP}{TC}} \right)^{.38} \quad (51)$$





FIGURE XIV

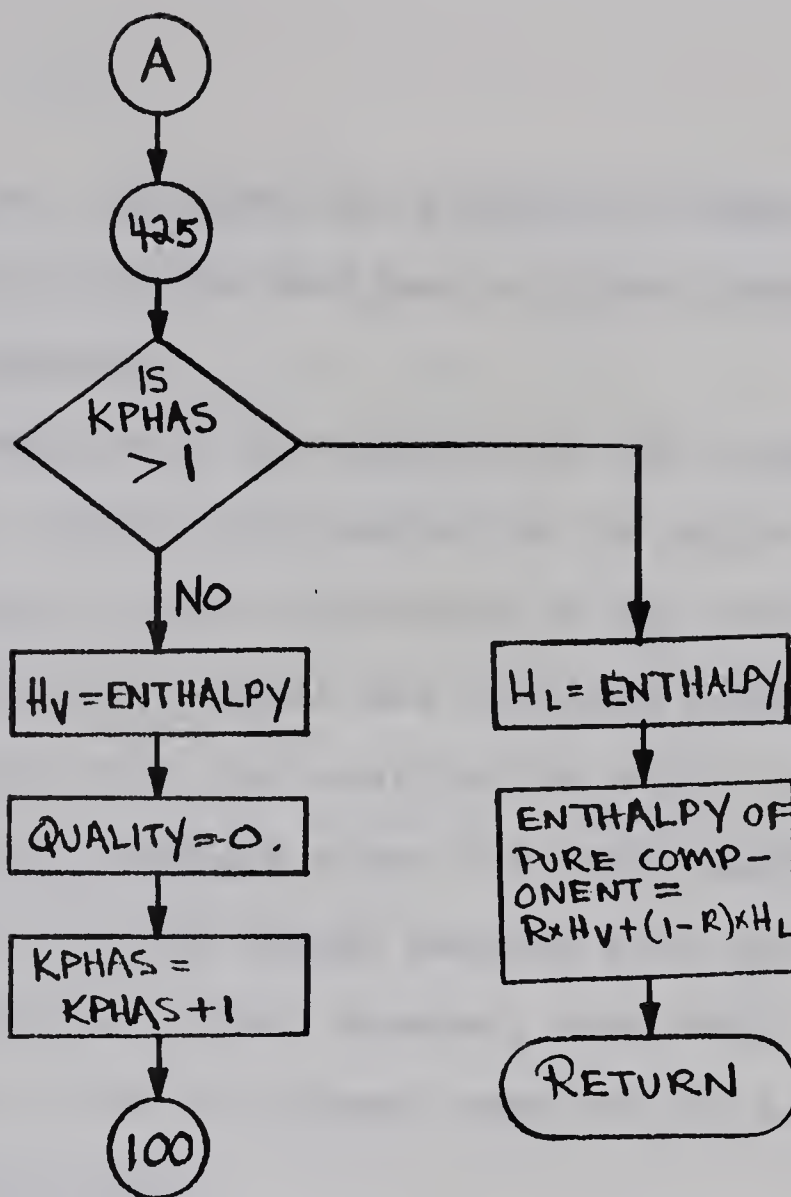
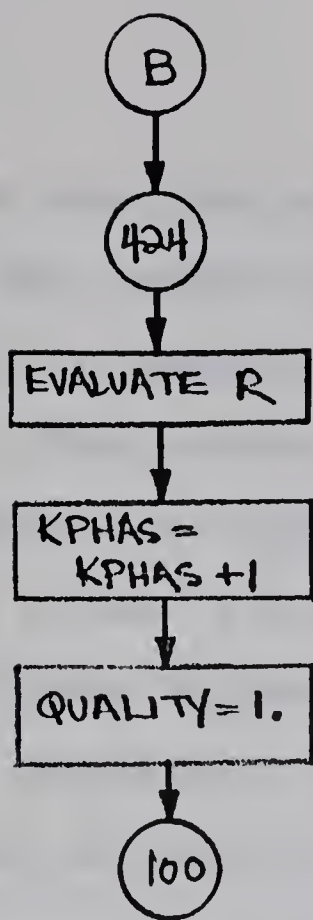
Flow Diagram for  
Subroutine ENTHPY



... CONTINUED









It can be seen from equation (51) that the reference temperature, TREF, should be less than the smallest critical temperature of the components involved.

The pressure, temperature and quality of the stream must be known (or assumed) before this subroutine is called. Ideal 'K values' are employed in the evaluation of the enthalpy of a two-phase stream. The method again was outlined previously. The only requirement here was that the equilibrium ratios are a function of temperature and pressure alone for each component. This maintains the linearity of the energy balance with respect to the mass flow variables. Note, however, that this concentration independence of the 'K values' need not be adhered to for any other subroutines.

#### 6. Subroutine BUBPT

In this work bubble point calculations are effected using ideal equilibrium ratios.

$$K = \frac{\text{Vapor Pressure}}{\text{Total Pressure}}$$

However, the subroutine for calculation of 'K values' could be replaced by any other suitable routine, for example, Chao-Seader equilibrium ratios or polynomial fit equilibrium ratios.

The method of convergence used is the same as that used to evaluate the unknown temperature in an energy balance, the method of false position.





At the bubble point temperature and pressure, the following relationship holds true where  $x_j$  and  $y_j$  represent the mole fractions of component  $j$  in the vapor and in the liquid states:

$$\sum_j y_j = 1.0$$

$$1.0 = \sum_j K_j x_j \quad (52)$$

Therefore, the problem becomes that of finding a temperature (given the pressure of the stream) such that:

$$\sum_j K_j x_j - 1.0 < \epsilon \quad (53)$$

where

$\epsilon$  is some specified error.

Two initial guesses are made at the bubble point temperature. Then the following equation is used thereafter to determine the temperature for the next iteration:

$$T^{(n+1)} = T^{(n)} - \frac{F(T^{(n)}) (T^{(n)} - T^{(n-1)})}{F(T^{(n)}) - F(T^{(n-1)})} \quad (54)$$

where

$$F(T^{(n)}) = \sum_{j=1}^{\text{NOCOMP}} (K_j)_{T^{(n)}} x_j - 1.0 \quad (55)$$



## 7. Subroutine DEWPT

The dew point calculation is based upon the following equation at the dew point:

$$\sum x_j = 1.0 = \sum \frac{y_j}{K_j} \quad (56)$$

The approach used is identical to that employed in the bubble point calculation.

## 8. Subroutine QUAL

This subroutine simply calculates the vapor ratio,  $V$ , and the quality,  $Q$ , of a stream given its composition, temperature and pressure. Quality is stored in the stream control variables vector in the fifth element,  $SNC(NSN,5)$  and the vapor ratio,  $V$ , is stored in  $SNC(NSN,8)$ . If the temperature of the stream is greater than the dew point temperature or less than the bubble point temperature, the quality is set to zero or one respectively. A flash calculation is required otherwise.

$V$  is the ratio of the pound moles of vapor to the per pound moles of mixture. The feed is one pound mole. From a component 'j' molal balance (Figure XV):

$$z_j = y_j V + x_j L \quad (57)$$

Defining:

$x_j$  = mole fraction of component  $j$  in the liquid phase of the two phase stream



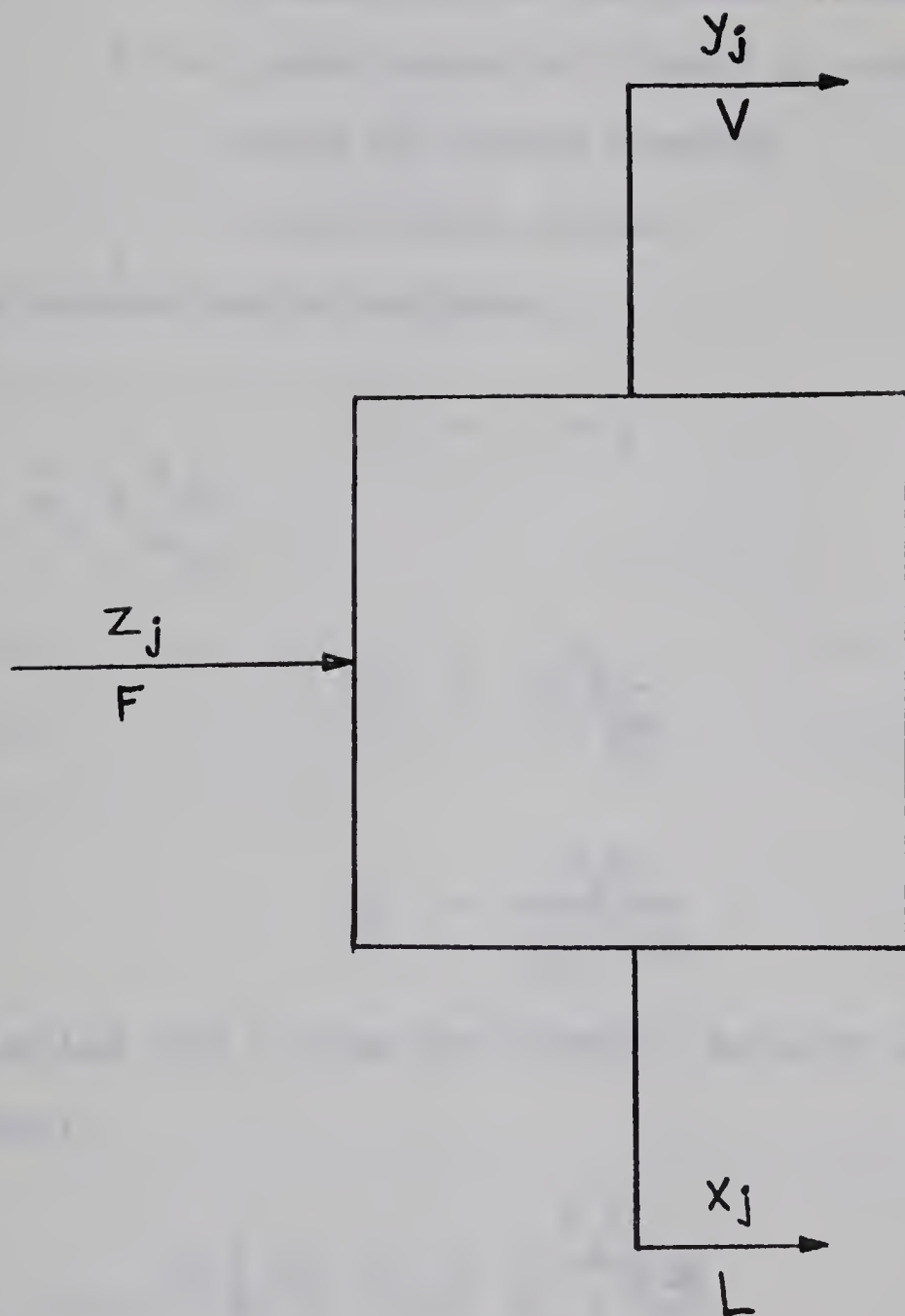


Figure XV

Two Phase Flow in Stream





$z_j$  = mole fraction of component  $j$  in the stream

$y_j$  = mole fraction of component  $j$  in the vapor phase of the two phase stream

$L$  = pound moles of liquid in stream per pound mole of stream mixture

$K_j$  = equilibrium ratio.

From an overall molal balance:

$$1 = V + L \quad (58)$$

Since  $x_j = \frac{y_j}{K_j}$

$$y_j = \frac{z_j}{V + \frac{L}{K_j}} \quad (59)$$

or

$$y_j = \frac{K_j z_j}{K_j V + L} \quad (60)$$

Substituting for  $L$  from the overall balance equation and rearranging:

$$V \sum_j y_j = \sum_j \frac{K_j z_j}{K_j + \frac{1-V}{V}} \quad (61)$$

or

$$V = \sum_j \frac{K_j z_j}{K_j + \frac{1-V}{V}} \quad (62)$$

or

$$1 = \sum_j \frac{K_j z_j}{V(K_j - 1) + 1} \quad (63)$$



Defining:

$$f(V) = \sum_j \frac{K_j z_j}{V(K_j - 1) + 1} - 1 \quad (64)$$

$$f'(V) = \frac{df(V)}{dV} = \sum_j \frac{-K_j z_j (K_j - 1)}{(V(K_j - 1) + 1)^2} \quad (65)$$

V is found by using the Newton-Raphson technique for finding the root of an equation:

$$V^{(n+1)} = V^{(n)} - \frac{f(V^{(n)})}{f'(V^{(n)})} \quad (66)$$

The initial guess for V is zero. As indicated previously V is stored in SNC(NSN,8) for two phase streams. The quality can now be evaluated:

$$Q = \frac{V \sum_i y_i M_i}{\sum_i z_i M_i} \quad (67)$$

where

$M_i$  is the molecular weight of component i.

## 9. Subroutine EQUILK

This subroutine calculates the equilibrium ratio of component I given temperature T and pressure P. Using constant values for the latent heat of vaporization the vapor pressure is evaluated from the Clausius-Clapeyron equation. The ideal equilibrium ratio is then obtained directly:

$$K_I = \frac{VP_I}{P} \quad (68)$$





As concluded in previous sections, any other method for calculating equilibrium ratios could be used here. The only requirement is that when calculating the enthalpy of a two phase stream the equilibrium ratios that are independent of concentration be used. Thus, the linearity of the energy balance with respect to the mass flow variables is maintained. The introduction of concentration dependence would result in non-linearity with respect to concentration variables in the energy balance for two phase flow and hence require a procedure for solving a non-linear set of equations.



#### IV. EQUIPMENT SUBROUTINES

By definition an equipment takes one or more input streams and performs some operation to produce one or more output streams. The equipment subroutine performs a mathematical operation to produce the output stream values. If all inputs to a unit are known or calculated then subroutine EQUIP does the bookkeeping for the unit calculation. SETUP is the bookkeeper for units involved in a trial and error calculation of a recycle loop. Reference is made to Mosler's thesis(9) for detailed information concerning the use and format of the equipment subroutine.

Information associated with the equipment subroutine is stored in STRMI, STRMCI, and STRMO and STRMCO matrices by the executive program. EQCALL calls the appropriate subroutine. The subroutine performs the calculations storing calculated output stream vectors in predetermined locations. These newly calculated stream vectors appear in the STRMO and STRMCO matrices when control is returned to the executive program.

The subroutine must calculate all the output mass flow variables and, if the energy balance of the process is to be completed, the temperatures and pressures of the output streams.

There are no restrictions imposed upon the equipment



subroutine other than the output stream storage. However, since the unit may be calculated many times if it is involved in a recycle loop, it is desirable to avoid rigorous numerical solutions in the interest of economizing on computer time.





## V. FUTURE WORK

The integrated program, although it represents a working executive program, can be extended and improved in several areas.

The incorporation of the unit calculation into the newly acquired material and energy balance portion of the executive program to supply additional relationships would be a desirable extension to this work. For example, when a reactor is included in a balance envelope the generation, represented by 'dummy' streams, contributes more unknowns to the equations if not specified. In some instances these 'dummy' stream variables may be expressed in terms of known quantities or other unknowns already part of the set of balance equations via the equations involved in the unit calculation.

Process data just as it is taken is not usually consistent. A consistent set of data is one for which material and energy balances close exactly. When data from existing plants is to be analyzed, there will be a necessity for subroutines to detect gross errors in the data and to adjust measurements to close the material and energy balances. Also, when a problem is over specified using plant data, a subroutine for solving a set equations in which the number of equations exceeds the number of unknowns using a criterion



such as least squares is desirable. The present material and energy balance routine as written searches only for equations equal in number to the number of unknowns involved.

This presentation assumes ideal solution behavior for calculating enthalpies and Raoult's law for equilibrium ratios. More sophisticated subroutines are desirable but they will require larger amounts of physical property data. Thus, the acquisition of physical property data is a problem which will have to be dealt with. The method used in this work is that of supplying physical property data as part of the data deck. However, this method would soon become impractical with the improvement of the subroutines mentioned above.

As discussed in the literature survey, the method of direct substitution employed by PACER requires research in the area of acceleration of convergence and of the determination of an optimum sequence of calculation of the units in the process. In fact, more work is demanded in the area of actual criteria for convergence of this method.

The power of optimization is the next step in building the PACER executive program. As it exists, a given process is evaluated for a given set of operating conditions. An optimization procedure could be treated as a unit calculation of a final unit in the process stream.

Concerning the application of the PACER program, the development of an equipment subroutine library is a necessity.







## VI. CONCLUSIONS

A working program has been developed which handles many of the typical problems involving the material and energy balance calculation of chemical processes. This program was formed by integrating a material and energy balance routine into an existing executive program, PACER. This balance routine performs material and energy balances around an arbitrary balance envelope involving one or more units.

A logic is included in the balance routine for finding an unknown input stream (either an inter-equipment or a feed stream) and then a feasible balance envelope to attempt a material and energy balance calculation. Balance envelopes introducing totally unknown streams are rejected. Information specified in product and in inter-equipment streams is exploited. This information was not employed by the PACER logic.

In the attempt to solve for the unknown mass flow variables associated with a balance envelope the routine tries single then simultaneous linear material and energy balances. However, an unknown temperature of a stream is calculated by solving the non-linear energy balance equation.

The logic of the balance routine automatically extends a balance envelope if the calculation of that envelope was unsuccessful.



PACER requires that each different unit be supplied with an equipment subroutine to calculate the outputs of that unit from the known inputs. The integrated program does not require an equipment subroutine for a unit calculation which consists of only a material and/or energy balance. The executive material and energy balance subroutines are available for this task. However, the material and energy balance routine may be followed by an equipment subroutine which then performs some other function such as an equipment design. This feature is illustrated in the examples appearing in the Appendices. Example I in Appendix A is solved by the material and energy balance routine alone and the PACER executive program simply calls the appropriate subroutine.

By calculating some inter-equipment streams through the material and energy balance logic the recycle calculation of a process can be eliminated in some cases, as exemplified in the problems presented in the Appendices.

If unknown feed streams can be calculated by the material and energy balance routine, then the rule of calculation inherent in PACER can be preserved (that is, the calculation of physical outputs from inputs as recommended for stability) and the calculations completed via the PACER logic. Problems involving feed streams to the process which are not totally specified are beyond the scope of many executive programs, including PACER. Several of the example problems in the





Appendices fall into this category.

Several subroutines included in the material and energy balance routine are available to the user:

1. The subroutine for calculation of the pure component enthalpy at the temperature and pressure of the stream is accessible to the user. This subroutine employs the Watson correlation and Van der Waals equation of state. When evaluating the enthalpy of a two-phase stream, equilibrium ratios which are independent of composition are used. Ideal equilibrium ratios using the Clausius Clapeyron equations were employed.
2. Subroutines for calculating the dew point and bubble point temperatures of a stream using equilibrium ratios are available.
3. A subroutine for calculating the quality and vapor ratio of a two-phase multi-component stream is also available.

All principal parts of the material and energy balance program were written as subroutines to facilitate substitution of new procedures selected by the user. For example, ideal 'K values' were employed but any other method of calculation could be used subject to the above restrictions.

The balance routine integrated with the modified PACER program was applied to the solution of the four example problems presented in the Appendices. Several different aspects of the integrated program are illustrated.





VII. BIBLIOGRAPHY

1. Aris, R., Nemhauser, G.L., and Wilde, D.J., A.I.Ch.E. Journal, 10, No. 6, p. 913 (1964).
2. Cavette, R.H., Proceedings Am. Pet. Inst., 43, No. 3, p. 57 (1963).
3. Dantos, S., Chem. Eng., 64, No. 10, p.284 (1957).
4. Fisher, D.G., Chemistry in Canada, June, p. 49 (1966).
5. Himmelblau, D.M., "Basic Principles and Calculations in Chemical Engineering", Prentice-Hall, Englewood Cliffs, N.J. (1962).
6. Hougen, O.A., Watson, R.M., and Ragatz, R.A., "Chemical Process Principles", Part II, John Wiley and Sons, New York (1962).
7. Kammermeyer, K., and Osburn, J., "Process Calculations", Prentice-Hall, Englewood Cliffs, N.J. (1956).
8. Kesler, M.G., and Griffiths, P.R., Proceedings Am. Pet. Inst., 43, No. 3, p. 49 (1963).
9. Mosler, H.A., "A M.Sc. Thesis Submitted to Purdue University," January, 1964.
10. Nagiev, M.F., Chem. Eng. Progr., 53, No. 10, p. 297 (1957).
11. Nagiev, M.F., "The Theory of Recycle Process in Chemical Engineering", MacMillan, New York (1964).
12. Naphtali, L.M., Chem. Eng. Progr., 60, No. 9, p. 70 (1964).
13. Norman, R.L., A.I.Ch.E. Journal, 11, No. 3, p. 40 (1965).
14. Ravicz, I.H., and Norman, R.L., Chem. Eng. Progr., 60, No. 5, p. 71 (1964).
15. Rinard, L.H., and Ripps, D.L., Chem. Eng. Progr. Symposium Series, 55, No. 61, p. 34 (1965).
16. Ripps, D.L., Chem. Eng. Progr. Symposium Series, 55, No. 61, p. 8 (1965).
17. Rosen, E.M., Chem. Eng. Progr., 58, No. 10, p. 69 (1962).



18. Rubin, D.I., Chem. Eng. Progr. Symposium Series, 37, No. 58, p. 54 (1962).
19. Rudd, D.F., and Lee, W., A.I.Ch.E. Journal, 12, No. 6, p. 1184 (1966).
20. Rudd, D.F., Lee, W., and Christensen, J.H., A.I.Ch.E. Journal, 12, No. 6, p. 1104, (1966).
21. Sargent, R.W.H., and Westerberg, A.W., Trans. Instn. Chem. Engrs., 42, p. T190, (1964).
22. Shannon, P.T., Chem. Eng. Ed., March, (1963).
23. Shannon, P.T., et al, "A Study of a Contact Sulphuric Acid Plant", McMaster University (1965).
24. Schmidt, A.X., and List, H.L., "Material and Energy Balances", Prentice-Hall, Englewood Cliffs, N.J., (1962).
25. Vela, M.A., Pet. Refiner, 40, No. 5, p. 247 (1962).
26. Vela, M.A., Pet. Refiner, 40, No. 6, p. 189 (1962).
27. Williams, E.T., and Johnson, C.R., "Stoichiometry for Chemical Engineers", McGraw-Hill, Toronto, (1958).









APPENDIX A

Example I

Reference: "Process Calculations"; Kammermeyer and Osburn;  
page 163; problem 5

Flow Diagram: See Figure A-1

Statement: The simplified flowsheet (Figure A-1) represents a section from a plant which manufactures butadiene ( $\text{H}_2\text{C} = \text{CH} - \text{CH} = \text{CH}_2$ ) from ethyl alcohol. The analysis of the labelled streams are given in Table A-1 in percent by weight. Since PACER is dimensioned to handle only eight components,  $\text{C}_2\text{H}_4\text{O}$  and  $\text{C}_4\text{H}_{10}\text{O}$  are combined and considered to be a single component. So are  $\text{CO}_2$  and  $\text{H}_2$ . These pairs occur together.

Using a basis of 10,000 lb/hr of stream 3, calculate the

1. mass flow rate of each stream
2. mass flow rate of water added in stream 15
3. mass flow rate of water removed in stream 12
4. recycle ratio (stream 2/stream 1)
5. recycle ratio (stream 11/stream 3)



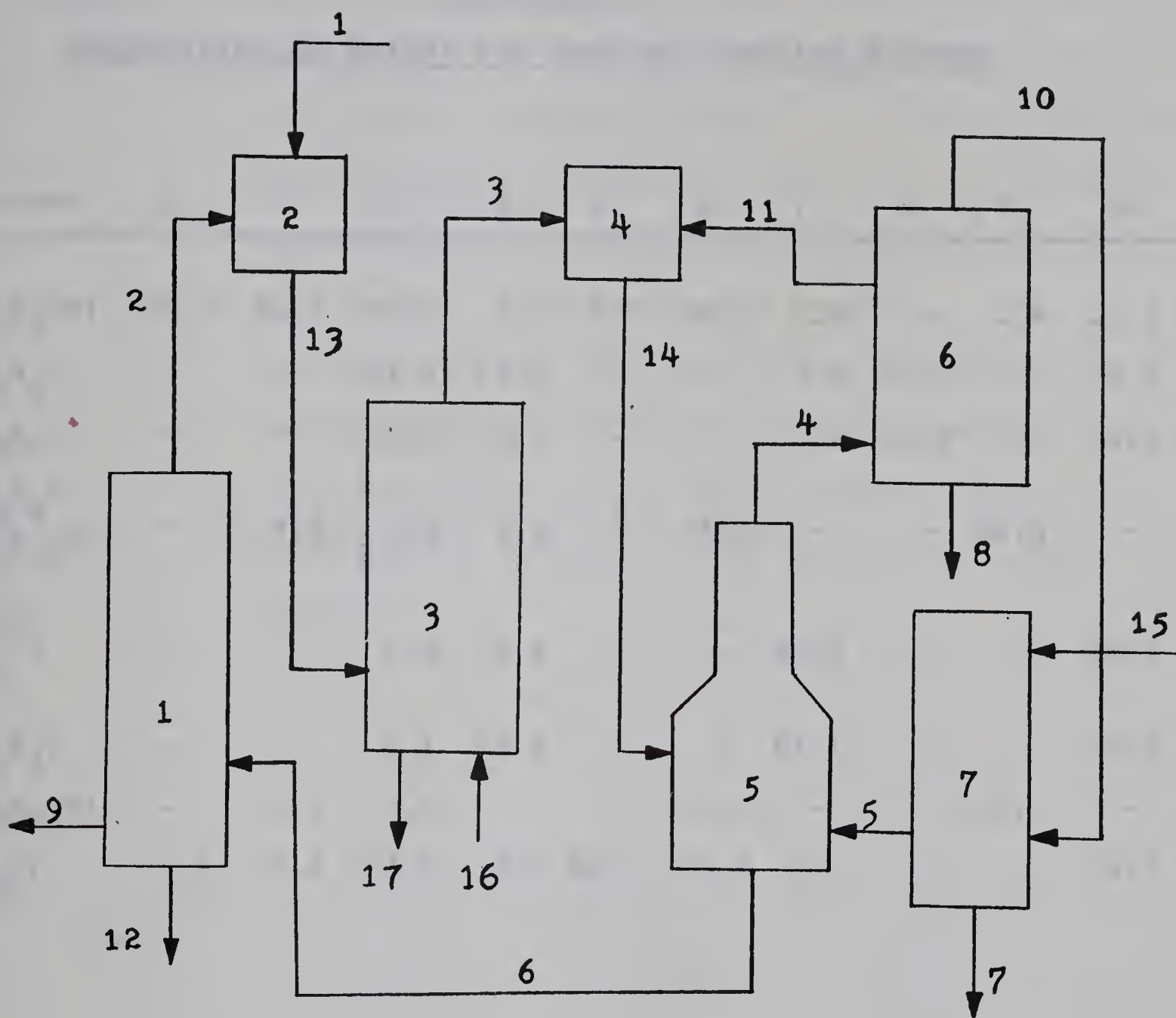


Figure A-1

Flowsheet of a Butadiene Plant

| UNIT | FUNCTION                |
|------|-------------------------|
| 1    | alcohol recovery        |
| 2    | mixer                   |
| 3    | reactor                 |
| 4    | mixer                   |
| 5    | scrubber                |
| 6    | absorbers and strippers |
| 7    | scrubber                |





TABLE A-1

Composition in Weight Per Cent of Labelled Streams

| Component  | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   | 11   |
|--|------|------|------|------|------|------|------|------|------|------|------|
| 1 (C <sub>2</sub> H <sub>5</sub> OH)                                       | 95.0 | 81.7 | 39.8 | 2.3  | 0.7  | 20.5 | 0.4  | -    | 2.6  | 11.1 | -    |
| 2 (C <sub>4</sub> H <sub>6</sub> )   | -    | -    | 14.8 | 60.0 | -    | -    | 3.6  | 80.0 | -    | 3.2  | 64.0 |
| 3 (C <sub>4</sub> H <sub>8</sub> )   | -    | -    | 3.7  | 15.0 | -    | -    | 0.9  | 20.0 | -    | 0.8  | 16.0 |
| 4 {<br>C <sub>2</sub> H <sub>4</sub> O<br>C <sub>4</sub> H <sub>10</sub> O | -    | 9.6  | 8.3  | 1.6  | -    | 4.3  | -    | -    | 64.4 | -    | 5.8  |
| 5 {<br>CO <sub>2</sub><br>H <sub>2</sub>                                   | -    | -    | 1.9  | 5.4  | -    | -    | 29.3 | -    | -    | 26.7 | -    |
| 6 (C <sub>2</sub> H <sub>4</sub> )   | -    | -    | 4.1  | 15.6 | -    | -    | 63.1 | -    | -    | 57.5 | 14.2 |
| 7 (C <sub>6</sub> H <sub>14</sub> O)                                       | -    | 1.3  | 2.5  | -    | -    | 1.3  | -    | -    | 33.  | -    | -    |
| 8 (H <sub>2</sub> O)   | 5.0  | 7.4  | 24.9 | 0.1  | 99.3 | 73.9 | 2.7  | -    | -    | 0.7  | -    |



Purpose: This example illustrates the balance routine searching for a suitable envelope around which to perform a material balance in order to calculate unknown inputs. The PACER logic for calculating equipment output streams or recycle loops is not used in the solution of this example. Although recycle is involved no trial and error calculations are necessary. No feed streams are totally specified and also no equipment sub-routines are required in the solution of this problem.





TABLE A-2

SAMPLE OF DATA FOR EXAMPLE I

SECTION OF BUTADIENE PLANT - KAMMERMEYER - PAGE 163 - PROBLEM 5

|   |   |   |    |   |   |    |
|---|---|---|----|---|---|----|
| 3 | 1 | 0 | 20 | 1 | 1 | 0  |
| 7 | 5 | 7 | 17 | 5 | 5 | 20 |
|   |   |   |    | 5 |   | 4  |

7

PROCESS MATRIX

|   |     |    |    |     |     |
|---|-----|----|----|-----|-----|
| 1 | 501 | 6  | -2 | -9  | -12 |
| 2 | 502 | 1  | 2  | -13 | -0  |
| 3 | 503 | 13 | 16 | -3  | -17 |
| 4 | 504 | 3  | 11 | -14 | -0  |
| 5 | 505 | 14 | 5  | -4  | -6  |
| 6 | 506 | 4  | -8 | -10 | -11 |
| 7 | 507 | 10 | 15 | -5  | -7  |

7

EQUIPMENT PARAMETERS MATRIX

|        |         |         |         |         |
|--------|---------|---------|---------|---------|
| 1.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 2.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 3.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 4.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 5.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 6.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 7.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |

7

EQUIPMENT CONTROL PARAMETERS MATRIX

|        |         |         |         |         |
|--------|---------|---------|---------|---------|
| 1.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 2.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 3.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 4.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 5.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 6.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |
| 7.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000 |

...Continued



## STREAM VARIABLES MATRIX

|          |           |            |           |          |
|----------|-----------|------------|-----------|----------|
| 1.0000   | 1.0000    | -1.0000    | 0.9500    | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | 0.0000    | 0.0000   |
| 0.0500   | -1.0000   | 0.0000     | 0.0000    | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | -1.0000   | 0.0000   |
| 2.0000   | 0.0000    | -1.0000    | 0.8170    | 0.0000   |
| 0.0000   | 0.0960    | 0.0000     | 0.0000    | 0.0130   |
| 0.0740   | -1.0000   | 0.0000     | 0.0000    | -1.0000  |
| 0.0000   | 0.0000    | -1.0000    | -1.0000   | 0.0000   |
| 3.0000   | 0.0000    | 10000.0000 | 0.3980    | 0.1480   |
| 0.0370   | 0.0830    | 0.0190     | 0.0410    | 0.0250   |
| 0.2490   | 3980.0000 | 1480.0000  | 370.0000  | 830.0000 |
| 190.0000 | 410.0000  | 250.0000   | 2490.0000 | 0.0000   |
| 4.0000   | 0.0000    | -1.0000    | 0.0230    | 0.6000   |
| 0.1500   | 0.0160    | 0.0540     | 0.1560    | 0.0000   |
| 0.0010   | -1.0000   | -1.0000    | -1.0000   | -1.0000  |
| -1.0000  | -1.0000   | -1.0000    | -1.0000   | 0.0000   |
| 5.0000   | 0.0000    | -1.0000    | 0.0070    | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | 0.0000    | 0.0000   |
| 0.9930   | -1.0000   | 0.0000     | 0.0000    | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | -1.0000   | 0.0000   |
| 6.0000   | 0.0000    | -1.0000    | 0.2050    | 0.0000   |
| 0.0000   | 0.0430    | 0.0000     | 0.0000    | 0.0130   |
| 0.7390   | -1.0000   | 0.0000     | 0.0000    | -1.0000  |
| 0.0000   | 0.0000    | -1.0000    | -1.0000   | 0.0000   |
| 7.0000   | 2.0000    | -1.0000    | 0.0040    | 0.0360   |
| 0.0090   | 0.0000    | 0.2930     | 0.6310    | 0.0000   |
| 0.0270   | -1.0000   | -1.0000    | -1.0000   | 0.0000   |
| -1.0000  | -1.0000   | 0.0000     | -1.0000   | 0.0000   |
| 8.0000   | 2.0000    | -1.0000    | 0.0000    | 0.8000   |
| 0.2000   | 0.0000    | 0.0000     | 0.0000    | 0.0000   |
| 0.0000   | 0.0000    | -1.0000    | -1.0000   | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | 0.0000    | 0.0000   |
| 9.0000   | 2.0000    | -1.0000    | 0.0260    | 0.0000   |
| 0.0000   | 0.6440    | 0.0000     | 0.0000    | 0.3300   |
| 0.0000   | -1.0000   | 0.0000     | 0.0000    | -1.0000  |
| 0.0000   | 0.0000    | -1.0000    | 0.0000    | 0.0000   |

...Continued





|         |         |         |         |            |
|---------|---------|---------|---------|------------|
| 10.0000 | 0.0000  | -1.0000 | 0.1110  | 0.0320     |
| 0.0080  | 0.0000  | 0.2670  | 0.5750  | 0.0000     |
| 0.0070  | -1.0000 | -1.0000 | -1.0000 | 0.0000     |
| -1.0000 | -1.0000 | 0.0000  | -1.0000 | 0.0000     |
| 11.0000 | 0.0000  | -1.0000 | 0.0000  | 0.6400     |
| 0.1600  | 0.0580  | 0.0000  | 0.1420  | 0.0000     |
| 0.0000  | 0.0000  | -1.0000 | -1.0000 | -1.0000    |
| 0.0000  | -1.0000 | 0.0000  | 0.0000  | 0.0000     |
| 12.0000 | 2.0000  | -1.0000 | 0.0000  | 0.0000     |
| 0.0000  | 0.0000  | 0.0000  | 0.0000  | 0.0000     |
| 1.0000  | 0.0000  | 0.0000  | 0.0000  | 0.0000     |
| 0.0000  | 0.0000  | 0.0000  | -1.0000 | 0.0000     |
| 13.0000 | 0.0000  | -1.0000 | -1.0000 | -1.0000    |
| -1.0000 | -1.0000 | -1.0000 | -1.0000 | -1.0000    |
| -1.0000 | -1.0000 | -1.0000 | -1.0000 | -1.0000    |
| -1.0000 | -1.0000 | -1.0000 | -1.0000 | -1.0000    |
| 14.0000 | 0.0000  | -0.0000 | -0.0000 | -0.0000    |
| -0.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000    |
| -0.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000    |
| -0.0000 | -0.0000 | -0.0000 | -0.0000 | -0.0000    |
| 15.0000 | 1.0000  | -1.0000 | 0.0000  | 0.0000     |
| 0.0000  | 0.0000  | 0.0000  | 0.0000  | 0.0000     |
| 1.0000  | 0.0000  | 0.0000  | 0.0000  | 0.0000     |
| 0.0000  | 0.0000  | 0.0000  | -1.0000 | 0.0000     |
| 16.0000 | 1.0000  | -1.0000 | 0.0000  | -1.0000    |
| -1.0000 | -1.0000 | -1.0000 | -1.0000 | -1.0000    |
| -1.0000 | 0.0000  | -1.0000 | -1.0000 | -1.0000    |
| -1.0000 | -1.0000 | -1.0000 | -1.0000 | 10000.0000 |
| 17.0000 | 2.0000  | -1.0000 | 1.0000  | 0.0000     |
| 0.0000  | 0.0000  | 0.0000  | 0.0000  | 0.0000     |
| 0.0000  | -1.0000 | 0.0000  | 0.0000  | 0.0000     |
| 0.0000  | 0.0000  | 0.0000  | 0.0000  | 10000.0000 |

...Continued





## STREAM CONTROL VARIABLES MATRIX

|         |        |         |        |         |
|---------|--------|---------|--------|---------|
| 1.0000  | 1.0000 | -0.0000 | 2.0000 | -0.0000 |
| 2.0000  | 0.0000 | -0.0000 | 4.0000 | -0.0000 |
| 3.0000  | 0.0000 | -0.0000 | 8.0000 | -0.0000 |
| 4.0000  | 0.0000 | -0.0000 | 7.0000 | -0.0000 |
| 5.0000  | 0.0000 | -0.0000 | 2.0000 | -0.0000 |
| 6.0000  | 0.0000 | -0.0000 | 4.0000 | -0.0000 |
| 7.0000  | 2.0000 | -0.0000 | 6.0000 | -0.0000 |
| 8.0000  | 2.0000 | -0.0000 | 2.0000 | -0.0000 |
| 9.0000  | 2.0000 | -0.0000 | 3.0000 | -0.0000 |
| 10.0000 | 0.0000 | -0.0000 | 6.0000 | -0.0000 |
| 11.0000 | 0.0000 | -0.0000 | 4.0000 | -0.0000 |
| 12.0000 | 2.0000 | -0.0000 | 1.0000 | -0.0000 |
| 13.0000 | 0.0000 | -0.0000 | 4.0000 | -0.0000 |
| 14.0000 | 0.0000 | -0.0000 | 8.0000 | -0.0000 |
| 15.0000 | 1.0000 | -0.0000 | 1.0000 | -0.0000 |
| 16.0000 | 1.0000 | -0.0000 | 6.0000 | -0.0000 |
| 17.0000 | 2.0000 | -0.0000 | 2.0000 | -0.0000 |

1

## STREAM VARIABLES TEST VECTOR

|          |          |        |        |        |
|----------|----------|--------|--------|--------|
| 100.0000 | 100.0000 | 0.0100 | 0.0100 | 0.0100 |
| 0.0100   | 0.0100   | 0.0100 | 0.0100 | 0.0100 |
| 0.0100   | 0.0100   | 0.0100 | 0.0100 | 0.0100 |
| 0.0100   | 0.0100   | 0.0100 | 0.0100 | 0.0100 |

1

## STREAM CONTROL VARIABLES TEST VECTOR

|          |          |          |          |        |
|----------|----------|----------|----------|--------|
| 100.0000 | 100.0000 | 100.0000 | 100.0000 | 0.5000 |
|----------|----------|----------|----------|--------|

0

8

0 -0.0000



The following discussion is presented to clarify the logic of the material and energy balance program. Reference is made to Figure VI and VII as well as Figure A-2.

Following the input routine (DREAD and DPRINT) control is transferred to subroutine MAJOR. All equipment and stream flags are initialized to zero. Subroutine SCAN is called. Totally known stream 3, see Figure A-2, is flagged to plus one and product streams 7, 8, 9, 12, and 17 to plus ten. Product streams do not affect the solution of the process. Totally known feed streams are flagged to plus seven by SCAN. However, neither feed stream 1 nor 15 is totally specified. Again refer to Figure A-2. Control returns to MAJOR.

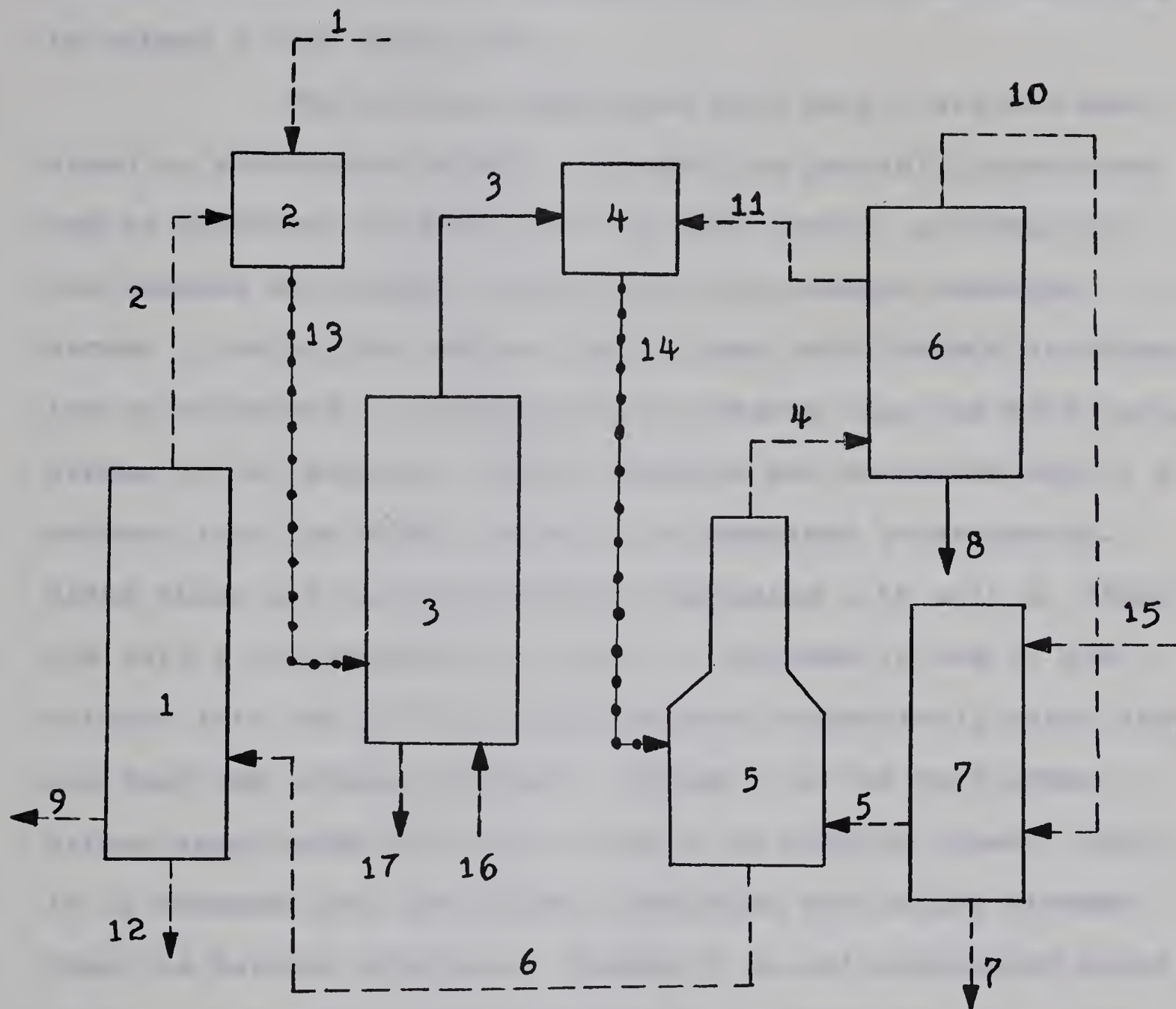
MAJOR scans the process matrix in an attempt to find a unit with all streams flowing into it flagged as known. There are none in this case as can be seen from a quick scan of Figure A-2.

Since no units can be calculated directly, subroutine MAJOR calls subroutine BALNCE in an attempt to calculate an unknown feed stream. The fresh feed streams are searched for unknowns by scanning the stream variables vectors which have a plus one in the second element of the vector (see Table A-2) signifying a feed stream. Feed stream 1 is not totally specified and therefore unit 2 is entered directly into the balance envelope. Note the negative numbers in









LEGEND

- TOTALLY KNOWN STREAM
- PARTIALLY KNOWN STREAM
- ..... UNKNOWN STREAM

Figure A-2

Flowsheet of Butadiene Plant  
Showing Status of Calculation



several elements of the stream variables matrix corresponding to stream 1 (see Table A-2).

The streams associated with unit 2 are now examined by subroutine BALNCE. Stream 2 is partially specified and is therefore entered into the NIPT vector, a vector of the numbers of streams flowing into the balance envelope. Stream 1, being the unknown feed stream which BALNCE is attempting to calculate, is automatically entered into the NIPT vector. Stream 13 is, however, totally unknown and therefore unit 3 is entered into the NUBAL vector to be examined subsequently. Since there are no more streams associated with unit 2, BALNCE now shifts its attention to unit 3. Streams 16 and 17 are entered into the NIPT and NOPT vectors respectively since they are feed and product streams. Stream 3 is the only other stream associated with unit 3 and it is totally known. Thus, it is entered into the vector containing the output streams from the balance envelope. Stream 13 is not considered since it flows between two units within the balance envelope. No new units were added to the NUBAL vector so a plausible balance envelope has been located. SEQCAL is called. Material balance calculations fail since the number of unknowns present exceed the number of equations available. The envelope is not extended since there are no partially unknown inter-equipment streams flowing from the balance envelope. Stream 3 is totally specified and nothing is to be gained by extending it. Refer





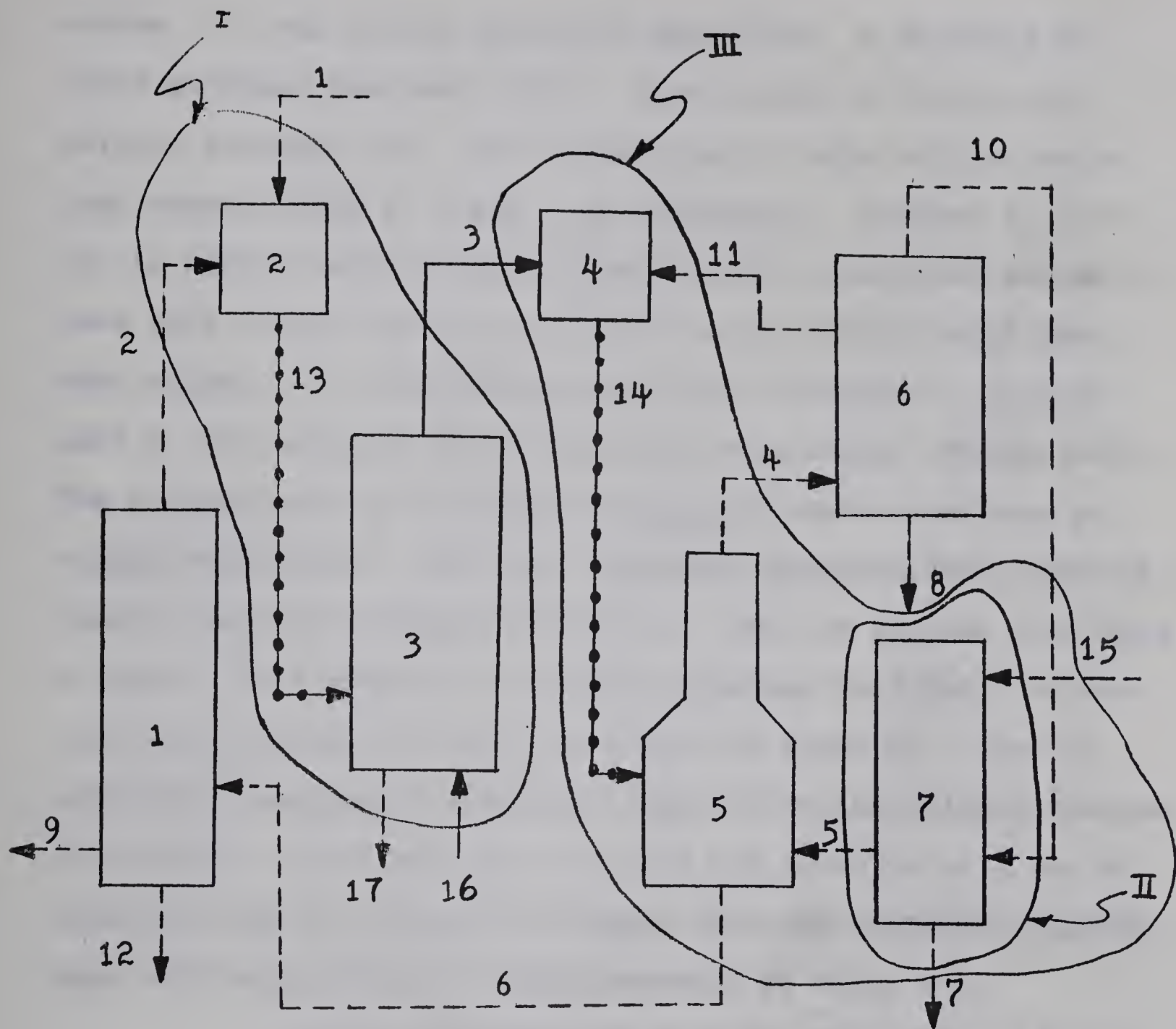
to Figure A-3 for illustrations showing the balance envelope attempted (balance envelope I).

Subroutine BALNCE attempts to extend the balance envelope first along unknown inter-equipment streams flowing from the balance envelope. Inter-equipment input streams from the initial balance envelope are not extended since the streams result from either recycle loops which will be uncovered when pursuing the output streams or from feed-forward loops in which case there is no need to repeat the balance envelope which has been or will be used in an attempt to calculate an unknown input stream upstream.

Scanning of the stream variables vectors is continued in search of another input stream which is not totally known. The next fresh feed stream which is not totally known is stream 15. Therefore, unit 7 becomes the starting point for the search for another balance envelope and the streams associated with unit 7 are examined. Since streams 5 and 10 are partially specified, a balance is attempted around unit 7. This is balance envelope II in Figure A-3. SEQCAL is called. Calculation of this balance envelope fails also. Control returns to BALNCE and attempts are made to extend this envelope. Stream 5 is extended since it is the only partially known inter-equipment stream which is an output from the balance. Thus, unit 5 is included in the balance envelope (see Figure A-3). Since stream 14 is totally unknown, unit 4 is also included in the balance envelope. Stream 3 is specified and







LEGEND  
SAME AS FIGURE A-2

Figure A-3

Flowsheet of Butadiene Plant  
Showing Status of Calculation

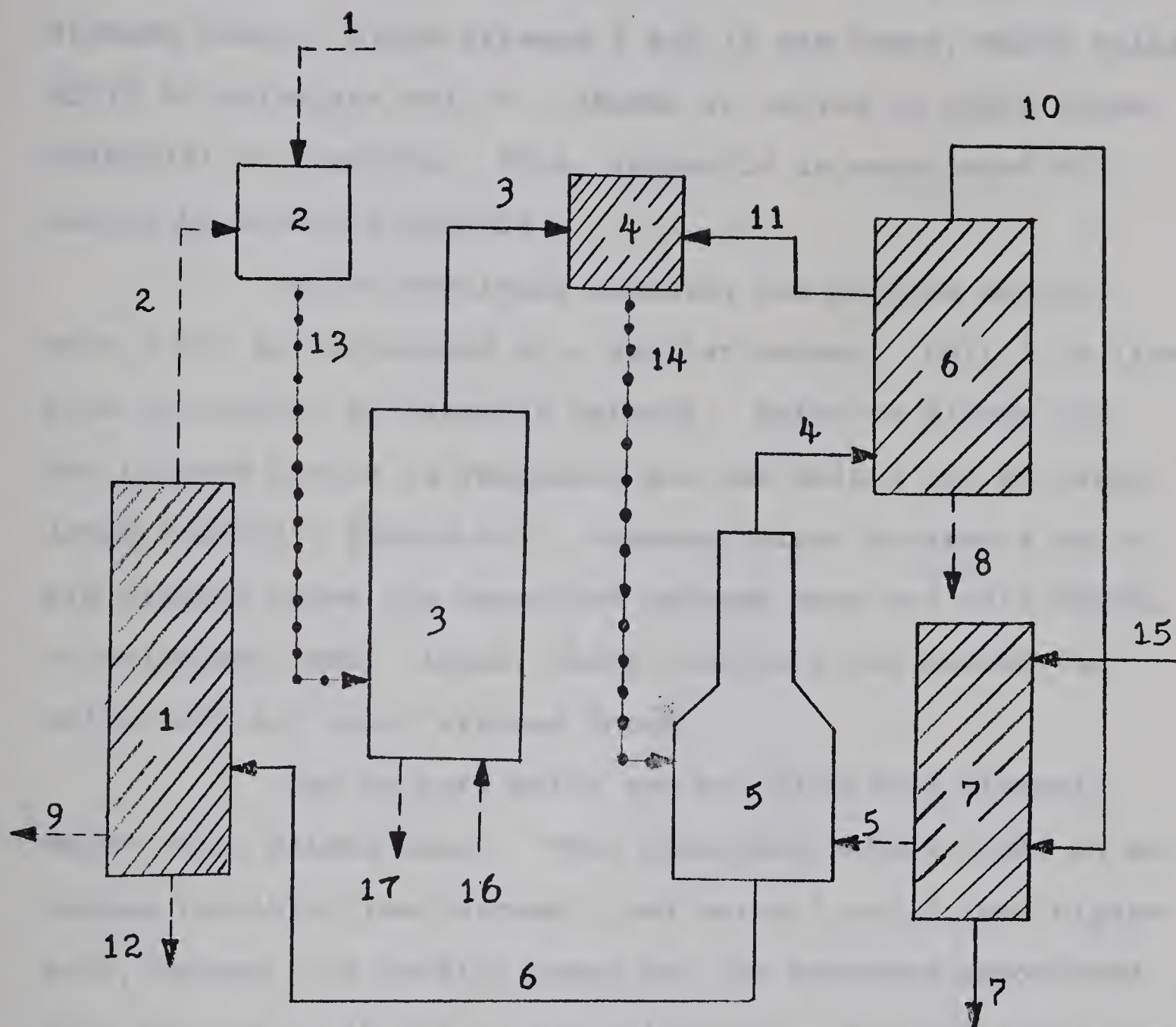


stream 4, 6 and 11 are partially specified. A suitable balance envelope has been found. Again refer to Figure A-3, balance envelope III. The calculation of this balance envelope around units 4, 5 and 7 is successful. Streams 4, 6, 7, 10, 11 and 15 are now known as well as the specified stream 3. Note this is not the only balance envelope which could have been solved. If this envelope had been extended to include unit 6, the solution would have been much easier (Figure A-3). The balance routine attempts the first plausible balance envelope encountered. Had this alternate envelope been used the results would be slightly different. This is because real data is used. This suggests a further extension to PACER. A routine which checks the data for errors is required. Also an additional subroutine should be supplied to the balance routine which uses a least squares criterion for solution of a set of equations which is larger in number than the unknowns involved. Some work along this line was presented by Ripps(16).

Figure A-4 shows the current status of the problem: known streams are shown by solid lines and the units for which all inputs are known are shaded. Referring to Figure A-4, all input streams to unit 1 are known and a balance around unit 1 allows the calculation of the unknown elements of the output stream variables vectors (stream 2, 9 and 12). This is accomplished by subroutine EQUIP calling SEQCAL since no equipment subroutine exists for unit 1 (see Figures VI and VII)







LEGEND

SAME AS FIGURE A-2

SHADED UNITS REPRESENT  
THOSE WHICH CAN BE  
CALCULATED DIRECTLY

Figure A-4

Flowsheet of Butadiene Plant

Showing Status of Calculation



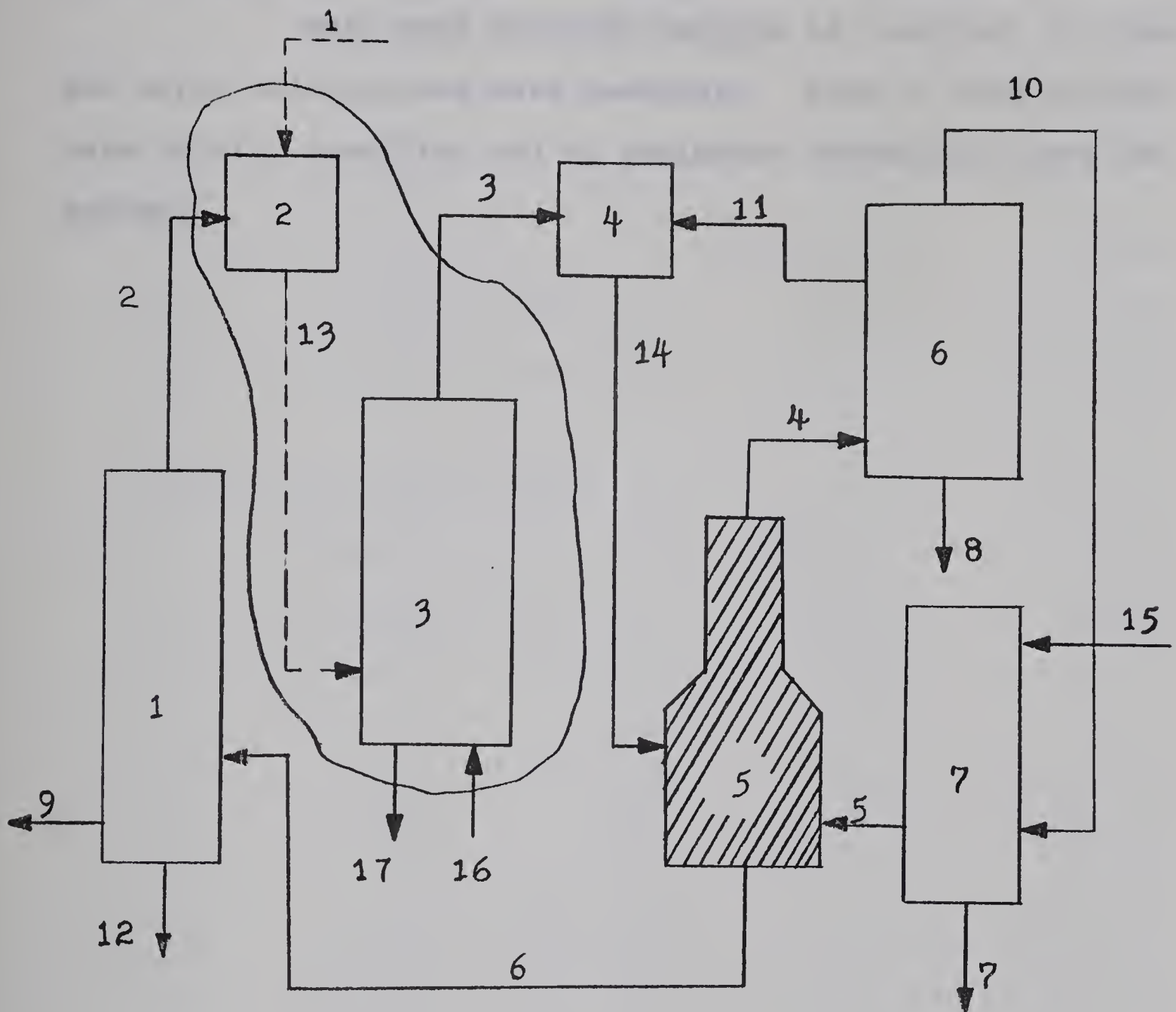
MAJOR continues searching for other units having all input streams known. Since streams 3 and 11 are known, MAJOR calls EQUIP to calculate unit 4. SEQCAL is called by EQUIP since NEXEQN(4) is positive. Thus, stream 14 is calculated directly by material balance.

MAJOR continues scanning the process matrix. Unit 6 can be calculated in a similar manner. Unit 7 is likewise calculated by material balance. Refer to Figure A-4. The process matrix is rescanned and now unit 5 can be calculated directly, Figure A-5. However, since streams 4 and 6 are already known the executive program does not call SEQCAL to calculate them. Again, MAJOR continues the search for units with all input streams known.

But no more units can be calculated directly. MAJOR calls BALNCE again. This time when BALNCE finds an envelope involving feed stream 1 and units 2 and 3 (see Figure A-5), stream 2 is totally known and the unknowns associated with streams 1, 16 and 17 are calculated. Control again returns to MAJOR. The process matrix is scanned and unit 2 is found to have all its input streams known. MAJOR calls EQUIP, which since NEXEQN(2) is positive, calls SEQCAL. Stream 13 is calculated directly by SEQCAL by material balance. Control returns to MAJOR and the scan of the process matrix continues. Now unit 3 has all its input streams known. MAJOR calls EQUIP. However, streams 3 and 17 are likewise known therefore SEQCAL need not be called. No more units are flagged as being un-







LEGEND

- SAME AS FIGURE A-2
- SHADED UNITS REPRESENT THOSE WHICH CAN BE CALCULATED DIRECTLY

Figure A-5

Flowsheet of Butadiene Plant  
Showing Status of Calculation





known therefore the process calculation is complete.

Note that although recycle is involved, no trial and error calculations were necessary. Also no feed streams were totally specified and no equipment subroutines were required.



TABLE A-3

Solution to Example I

(refer to Table A-4)

| The Recycle Ratios      | Kammermeyer | Program |
|-------------------------|-------------|---------|
| 1. stream 2/stream 1    | 0.94        | 0.933   |
| 2. stream 11/stream 3   | 0.102       | 0.0974  |
| Mass Flow Rate of Water |             |         |
| 1. added at 15          | 11,810.     | 11,796. |
| 2. removed at 12        | 13,924.     | 13,915. |





TABLE A-4

Stream Variables Matrix at Run 3

|          |           |            |            |          |
|----------|-----------|------------|------------|----------|
| 1.0000   | 1.0000    | 5172.3151  | 0.9500     | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | 0.0000     | 0.0000   |
| 0.0500   | 4913.6993 | 0.0000     | 0.0000     | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | 258.6157   | 0.0000   |
| 2.0000   | 0.0000    | 4827.6849  | 0.8170     | 0.0000   |
| 0.0000   | 0.0960    | 0.0000     | 0.0000     | 0.0130   |
| 0.0740   | 3944.2186 | 0.0000     | 0.0000     | 463.4578 |
| 0.0000   | 0.0000    | 62.7599    | 357.2487   | 0.0000   |
| 3.0000   | 0.0000    | 10000.0000 | 0.3980     | 0.1480   |
| 0.0370   | 0.0830    | 0.0190     | 0.0410     | 0.0250   |
| 0.2490   | 3980.0000 | 1480.0000  | 370.0000   | 830.0000 |
| 190.0000 | 410.0000  | 250.0000   | 2490.0000  | 0.0000   |
| 4.0000   | 0.0000    | 3505.0957  | 0.0230     | 0.6000   |
| 0.1500   | 0.0160    | 0.0540     | 0.1560     | 0.0000   |
| 0.0010   | 80.6172   | 2103.0574  | 525.7644   | 56.0815  |
| 189.2752 | 546.7949  | 0.0000     | 3.5051     | 0.0000   |
| 5.0000   | 0.0000    | 11843.0922 | 0.0070     | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | 0.0000     | 0.0000   |
| 0.9930   | 59.6515   | 0.0000     | 0.0000     | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | 11786.4011 | 0.0000   |
| 6.0000   | 0.0000    | 19312.3623 | 0.2050     | 0.0000   |
| 0.0000   | 0.0430    | 0.0000     | 0.0000     | 0.0130   |
| 0.7390   | 3959.0343 | 0.0000     | 0.0000     | 830.4316 |
| 0.0000   | 0.0000    | 251.0607   | 14271.8357 | 0.0000   |
| 7.0000   | 2.0000    | 508.8984   | 0.0040     | 0.0360   |
| 0.0090   | 0.0000    | 0.2930     | 0.6310     | 0.0000   |
| 0.0270   | 2.0356    | 18.3203    | 4.5801     | 0.0000   |
| 149.1072 | 321.1149  | 0.0000     | 13.7403    | 0.0000   |
| 8.0000   | 2.0000    | 1974.9907  | 0.0000     | 0.8000   |
| 0.2000   | 0.0000    | 0.0000     | 0.0000     | 0.0000   |
| 0.0000   | 0.0000    | 1461.6797  | 365.4199   | 0.0000   |
| 0.0000   | 0.0000    | 0.0000     | 0.0000     | 0.0000   |
| 9.0000   | 2.0000    | 569.8351   | 0.0260     | 0.0000   |
| 0.0000   | 0.6440    | 0.0000     | 0.0000     | 0.3300   |
| 0.0000   | 14.8157   | 0.0000     | 0.0000     | 366.9738 |
| 0.0000   | 0.0000    | 188.0456   | 0.0000     | 0.0000   |



- A21 -

|          |           |            |            |            |
|----------|-----------|------------|------------|------------|
| 10.0000  | 0.0000    | 555.7393   | 0.1110     | 0.0320     |
| 0.0080   | 0.0000    | 0.2670     | 0.5750     | 0.0000     |
| 0.0070   | 61.6871   | 17.7837    | 4.4459     | 0.0000     |
| 148.3824 | 319.5501  | 0.0000     | 3.8902     | 0.0000     |
| 11.0000  | 0.0000    | 974.3657   | 0.0000     | 0.6400     |
| 0.1600   | 0.0580    | 0.0000     | 0.1420     | 0.0000     |
| 0.0000   | 0.0000    | 623.5941   | 155.8985   | 56.5132    |
| 0.0000   | 138.3599  | 0.0000     | 0.0000     | 0.0000     |
| 12.0000  | 2.0000    | 13914.8422 | 0.0000     | 0.0000     |
| 0.0000   | 0.0000    | 0.0000     | 0.0000     | 0.0000     |
| 1.0000   | 0.0000    | 0.0000     | 0.0000     | 0.0000     |
| 0.0000   | 0.0000    | 0.0000     | 13914.8422 | 0.0000     |
| 13.0000  | 0.0000    | 10000.0000 | 0.8858     | -0.0000    |
| -0.0000  | 0.0463    | -0.0000    | -0.0000    | 0.0063     |
| 0.0616   | 8857.9178 | -0.0000    | -0.0000    | 463.4578   |
| -0.0000  | -0.0000   | 62.7599    | 615.8644   | -1.0000    |
| 14.0000  | 0.0000    | 10974.3657 | 0.3627     | 0.1917     |
| 0.0479   | 0.0808    | 0.0173     | 0.0500     | 0.0228     |
| 0.2269   | 3980.0000 | 2103.5941  | 525.8985   | 886.5132   |
| 190.0000 | 548.3599  | 250.0000   | 2490.0000  | -0.0000    |
| 15.0000  | 1.0000    | 11796.2513 | 0.0000     | 0.0000     |
| 0.0000   | 0.0000    | 0.0000     | 0.0000     | 0.0000     |
| 1.0000   | 0.0000    | 0.0000     | 0.0000     | 0.0000     |
| 0.0000   | 0.0000    | 0.0000     | 11796.2513 | 0.0000     |
| 16.0000  | 1.0000    | 4877.9178  | 0.0000     | 0.3034     |
| 0.0759   | 0.0751    | 0.0390     | 0.0841     | 0.0384     |
| 0.3842   | 0.0000    | 1480.0000  | 370.0000   | 366.5422   |
| 190.0000 | 410.0000  | 187.2401   | 1874.1356  | 10000.0000 |
| 17.0000  | 2.0000    | 4877.9178  | 1.0000     | 0.0000     |
| 0.0000   | 0.0000    | 0.0000     | 0.0000     | 0.0000     |
| 0.0000   | 4877.9178 | 0.0000     | 0.0000     | 0.0000     |
| 0.0000   | 0.0000    | 0.0000     | 0.0000     | 10000.0000 |





APPENDIX B

Example II

Reference: Ch.E. 562 (Mass Transfer) Problem Assignment  
Dr. F.D. Otto

Flow Diagram: Figure B-1

Statement: In a proposed process mixtures of n-butanol and methyl-n-butyl ketone must be separated. Since the mixture is reported to form an azeotrope, extraction with water is to be investigated. Liquid equilibria are provided by Jones and McCants (Ind. Eng. Chem., 46, 1965) at 100°F, Table B-2

A - ketone

B - water

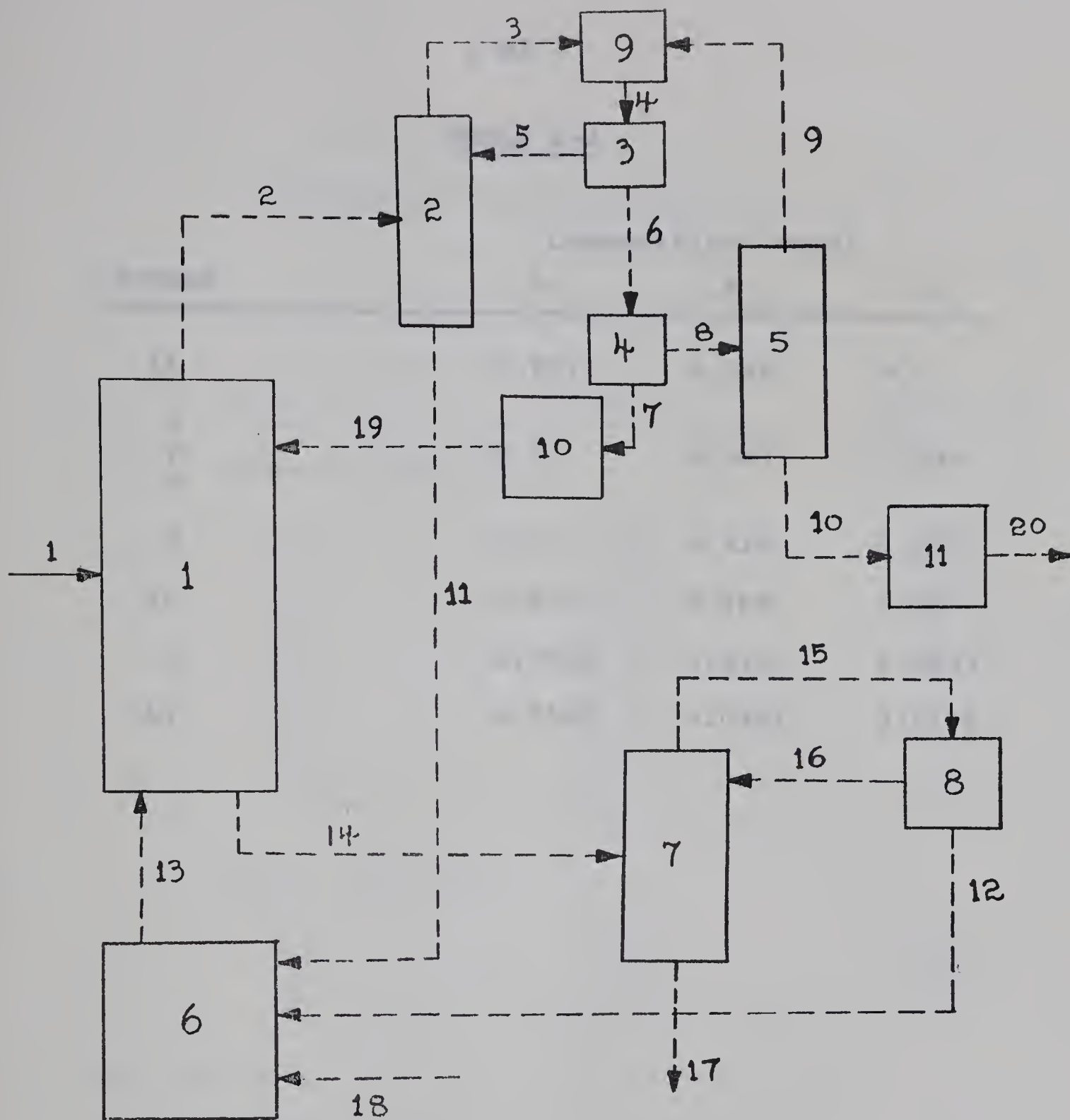
C - butanol

Refer to Table B-1 for stream compositions  
To obtain the composition of stream 14 a solvent free coordinate (water free) diagram was used. This composition was simply supplied to the program as part of the input data.

Note: Referring to Figure B-1, units 10 and 11 are pseudo-units introduced to conveniently handle the fact that the reflux ratio is specified. The subroutine for these units was the only







LEGEND

SAME AS FIGURE A-2

Figure B-1

Flow Diagram for Example II

| Unit | Function            | Unit | Function            |
|------|---------------------|------|---------------------|
| 1    | extractor           | 6    | solvent storage     |
| 2    | still               | 7    | still               |
| 3    | condenser-separator | 8    | condenser-separator |
| 4    | splitter            | 9    | mixer               |
| 5    | still               | 10   | pseudo unit         |
|      |                     | 11   | pseudo unit         |



TABLE B-1

| Stream | Composition (wt.%) |        |        |
|--------|--------------------|--------|--------|
|        | A                  | B      | C      |
| 16     | 0.971              | 0.029  | 0.0    |
| 6      |                    |        |        |
| 7      | 0.0                | 0.203  | 0.797  |
| 8      |                    |        |        |
| 5      | 0.0                | 0.928  | 0.072  |
| 12     | 0.015              | 0.985  | 0.00   |
| 2      | 0.0035             | 0.933  | 0.0635 |
| 14     | 0.9505             | 0.0301 | 0.0194 |





TABLE B-2

Equilibrium Data for Example II

1. Solubility Data (weight percent) at 100°F

BuOH = C                      Ketone = A                      Water = B

---

Organic Rich Curve

|      |      |      |
|------|------|------|
| 79.7 | 0.0  | 20.3 |
| 77.0 | 3.1  | 19.9 |
| 74.7 | 6.0  | 19.3 |
| 66.7 | 16.0 | 17.3 |
| 56.6 | 28.4 | 15.0 |
| 44.2 | 44.3 | 11.5 |
| 30.5 | 61.1 | 8.4  |
| 18.9 | 75.5 | 5.6  |
| 11.3 | 84.5 | 4.2  |
| 0.0  | 97.1 | 2.9  |

Water Rich Curve

|     |     |      |
|-----|-----|------|
| 0.0 | 1.5 | 98.5 |
| 3.6 | 1.0 | 95.4 |
| 7.2 | 0.0 | 92.8 |

2. Tie Line Data

Organic Phase

|      |      |      |
|------|------|------|
| 66.7 | 16.0 | 17.3 |
| 55.0 | 30.6 | 14.4 |
| 37.5 | 52.0 | 10.5 |
| 11.3 | 84.6 | 4.1  |

Water Phase

|     |     |      |
|-----|-----|------|
| 6.4 | 0.3 | 93.3 |
| 5.1 | 0.6 | 94.3 |
| 3.6 | 1.0 | 95.4 |
| 1.5 | 1.5 | 97.0 |



equipment subroutine written for this example and it simply calculates the unknown flow rate in the recycle ratio when the other is specified.

No ternary azeotrope is reported.

Assume:

1. Distillation of extract (stream 2) yields butanol-water azeotrope (stream 3). The water-rich layer of condensate (stream 5) is returned as still reflux, while the butanol-rich layer provides extractor reflux (stream 7) and the stream 8. Stream 8 is rectified to give stream 9 (same composition as stream 3) and the extract product stream 10. Residues (stream 11) may be assumed to be free of butanol.
2. Raffinate stream 14, when rectified provides the ketone water azeotrope (stream 15), the ketone-rich layer (stream 16) of which is returned to the still while the water-rich layer (stream 12) is withdrawn. Residue (stream 17), raffinate product may be assumed to be free of water.
3. The azeotrope compositions are:  
stream 3 and 9 = 42.5% water  
stream 15 = 30% water
4. Streams 5, 7, 16, and 17 are saturated liquids.



The specifications of:

1. feed: 1000 lb/hr 50% ketone, 50% butanol
2. extract product: 2% water, 98% butanol
3. raffinate product: 2% butanol, 98% ketone
4. extract: 95% butanol, 5% ketone on water  
free basis

Extract reflux (stream 7/stream 10) is 2.0 (total weights).

Determine total rates of all streams.





TABLE B-3

Solution to Example II

| Stream | Mass Flow Rate (lb/hr)          |                  |
|--------|---------------------------------|------------------|
|        | Program<br>(refer to Table B-4) | Hand Calculation |
| 2      | 20259.3                         | 20270.           |
| 3      | 2343.0                          | 2347.            |
| 5      | 843.6                           | 847.             |
| 6      | 1911.4                          | 1893.            |
| 7      | 999.6                           | 1000.            |
| 8      | 911.8                           | 893.             |
| 10     | 499.8                           | 500.             |
| 11     | 18759.9                         | 18770.           |
| 12     | 15.8                            | 16.0             |
| 13     | 18785.7                         | 18796.           |
| 14     | 526.0                           | 526.0            |
| 15     | 55.7                            | 56.4             |
| 16     | 39.9                            | 40.4             |
| 17     | 510.2                           | 510.             |
| 18     | 9.996                           | 10.              |

| Stream Component |   | Composition |       |
|------------------|---|-------------|-------|
| 11               | 1 | 0.9962      | 0.996 |
|                  | 2 | 0.0038      | 0.004 |
|                  | 3 | 0.0         | 0.0   |
| 13               | 1 | 0.9962      | 0.996 |
|                  | 2 | 0.0038      | 0.004 |
|                  | 3 | 0.0         | 0.0   |



TABLE B-4

Stream Variables Matrix at Run 2

|         |            |            |           |        |
|---------|------------|------------|-----------|--------|
| 1.0000  | 1.0000     | 1000.0000  | 0.0000    | 0.5000 |
| 0.5000  | 0.0000     | 500.0000   | 500.0000  | 0.0000 |
| 2.0000  | 0.0000     | 20259.2747 | 0.9330    | 0.0035 |
| 0.0635  | 18901.9031 | 70.9075    | 1286.4640 | 0.0000 |
| 3.0000  | 0.0000     | 2342.9599  | 0.4250    | 0.0000 |
| 0.5750  | 995.7580   | 0.0000     | 1347.2019 | 0.0000 |
| 4.0000  | 0.0000     | 2754.9504  | 0.4250    | 0.0000 |
| 0.5750  | 1170.8539  | 0.0000     | 1584.0965 | 0.0000 |
| 5.0000  | 0.0000     | 843.5844   | 0.9280    | 0.0000 |
| 0.0720  | 782.8463   | 0.0000     | 60.7381   | 0.0000 |
| 6.0000  | 0.0000     | 1911.3647  | 0.2030    | 0.0000 |
| 0.7970  | 388.0070   | 0.0000     | 1523.3576 | 0.0000 |
| 7.0000  | 0.0000     | 999.5835   | 0.2030    | 0.0000 |
| 0.7970  | 202.9154   | 0.0000     | 796.6680  | 0.0000 |
| 8.0000  | 0.0000     | 911.7822   | 0.2030    | 0.0000 |
| 0.7970  | 185.0918   | 0.0000     | 726.6904  | 0.0000 |
| 9.0000  | 0.0000     | 411.9905   | 0.4250    | 0.0000 |
| 0.5750  | 175.0960   | 0.0000     | 236.8945  | 0.0000 |
| 10.0000 | 0.0000     | 499.7918   | 0.0200    | 0.0000 |
| 0.9800  | 9.9958     | 0.0000     | 489.7959  | 0.0000 |
| 11.0000 | 0.0000     | 18759.8992 | 0.9962    | 0.0038 |
| 0.0000  | 18688.9917 | 70.9075    | 0.0000    | 0.0000 |
| 12.0000 | 0.0000     | 15.7800    | 0.9850    | 0.0150 |
| 0.0000  | 15.5433    | 0.2367     | 0.0000    | 0.0000 |
| 13.0000 | 0.0000     | 18785.6748 | 0.9962    | 0.0038 |
| 0.0000  | 18714.5305 | 71.1442    | 0.0000    | 0.0000 |
| 14.0000 | 0.0000     | 525.9838   | 0.0301    | 0.9505 |
| 0.0194  | 15.8321    | 499.9476   | 10.2041   | 0.0000 |
| 15.0000 | 0.0000     | 55.6668    | 0.3000    | 0.7000 |
| 0.0000  | 16.7000    | 38.9668    | 0.0000    | 0.0000 |





|         |          |          |          |         |
|---------|----------|----------|----------|---------|
| 16.0000 | 0.0000   | 39.8868  | 0.0290   | 0.9710  |
| 0.0000  | 1.1567   | 38.7301  | -0.0000  | -0.0000 |
| 17.0000 | 2.0000   | 510.2041 | 0.0000   | 0.9800  |
| 0.0200  | 0.0000   | 500.0000 | 10.2041  | 0.0000  |
| 18.0000 | 1.0000   | 9.9958   | 1.0000   | 0.0000  |
| 0.0000  | 9.9958   | 0.0000   | 0.0000   | 0.0000  |
| 19.0000 | 0.0000   | 999.5835 | 0.2030   | 0.0000  |
| 0.7970  | 202.9154 | 0.0000   | 796.6680 | 0.0000  |
| 20.0000 | 2.0000   | 499.7918 | 0.0200   | 0.0000  |
| 0.9800  | 9.9958   | 0.0000   | 489.7959 | 0.0000  |



APPENDIX C

Example III

Reference: "Basic Principles and Calculations in Chemical Engineering"; Himmelblau; page 352; problem 5-14

Flow Diagram: Figure C-1

Statement: Grain mash is fed through a heat exchanger where it is heated to 170°F. The alcohol is removed as 60% alcohol from the first column. The bottoms contain no alcohol. The 60% alcohol is further fractionated to 95% alcohol and essentially pure water in the second column. Heat is supplied to the bottoms of the columns by steam. Condenser water is available at 80°F.

Feed specification (streams 1 and 2):

80% water  
10% ethanol  
10% organic material

- a) Determine the weight of the following streams in pounds per hour:
1. overhead product Column I
  2. reflux Column I
  3. bottoms Column I
  4. overhead product Column II
  5. reflux Column II
  6. bottoms Column II



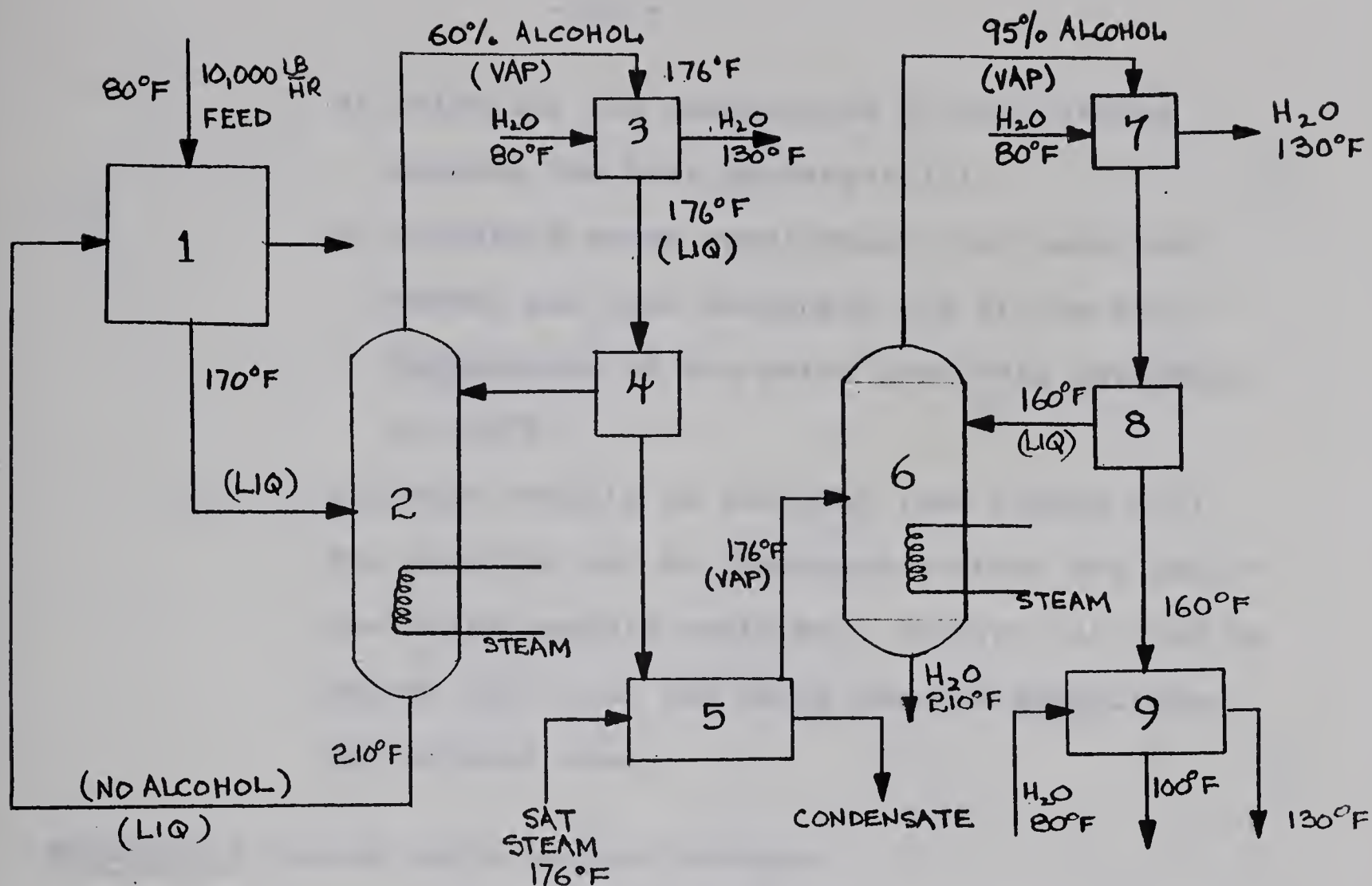


Figure C-1

Flow Diagram for Example III

| Unit | Function               |
|------|------------------------|
| 1    | Heat Exchanger III     |
| 2    | Column I (3/1 reflux)  |
| 3    | Condenser I            |
| 4    | Splitter               |
| 5    | Heat Exchanger I       |
| 6    | Column II (3/1 reflux) |
| 7    | Condenser II           |
| 8    | Splitter               |
| 9    | Heat Exchanger II      |





- b) Calculate the temperature of the bottoms leaving the heat exchanger III.
- c) Calculate water requirements for each condenser and heat exchanger III if the exit temperature of the water from this equipment is 130°F.

Although recycle is involved (see Figure C-1), the solution can be completed without the use of the PACER recycle routines. However, it will be solved both with the PACER recycle subroutines and without them.

#### Solution I (using PACER recycle routine)

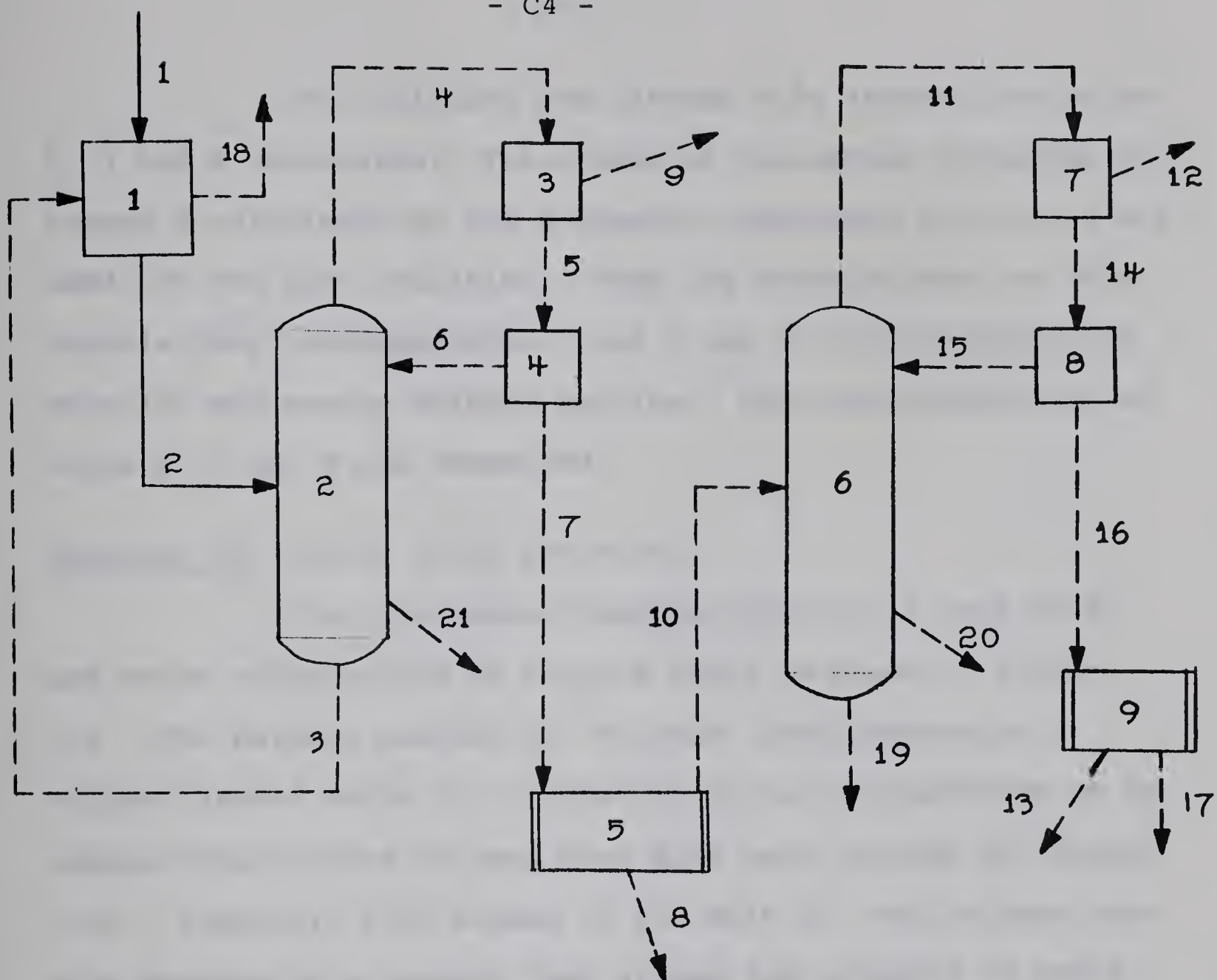
Referring to Figure C-2, the flowsheet is modified slightly to be adapted to PACER input.

An equipment subroutine is required to calculate the split attained in units 4 and 8. Also equipment subroutines are required to calculate the cooling water flow rates and steam rates after the balance routine calculated the heat loads (streams 8, 9, 12, and 13).

Trial and error via PACER recycle routines is required for units 2, 3 and 4 and units 6, 7 and 8. Units 1 and 2 also constitute a recycle loop, however, unit 1 can be calculated directly after the recycle calculation of units 2, 3 and 4 converge. Note that units 2, 3 and 4 could be considered as a single unit requiring one unit subroutine.



- C4 -



LEGEND

SAME AS FIGURE A-2

Figure C-2

Flowsheet for Solution I





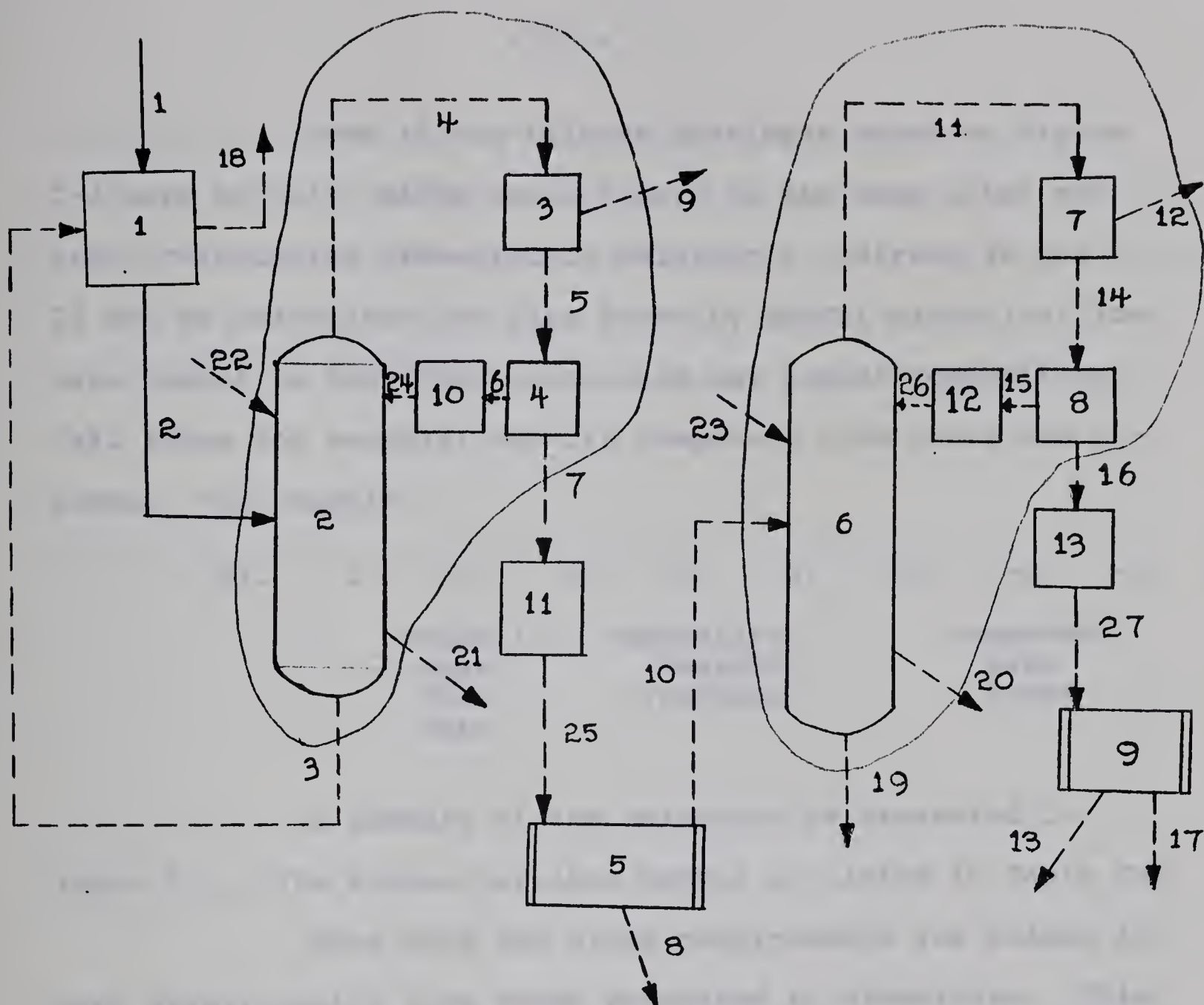
For solution one, stream 6 is assumed and units 2, 3 and 4 calculated. The values of the stream variables of stream 6 calculated by the equipment subroutine for unit 4 are used for the next iteration. When the calculations for this recycle loop converge units 1 and 5 can be calculated by the material and energy balance routine. Then the calculation of units 6, 7 and 8 are commenced.

Solution II (avoids trial and error)

The alternate flowsheet which will omit trial and error calculations of recycle loops is shown in Figure C-3. The balance routine is "tricked" into performing a balance around units 2, 3, 4 and 10 by the introduction of an unknown feed stream of zero mass flow rate (stream 22, Figure C-3). Similarly with stream 23 for unit 6. The balance routine detects this unknown feed stream and attempts to solve for it by performing a balance around units 2 and 10. This fails; however, upon its extension to include units 3 and 4, the balance routine is able to solve for unknowns in streams 3 and 7 as well as 22. The routine is not able to solve the energy balance. That is, heat flows (streams 2 and 9) remain unknown.

Pseudo-units(10,11,12,13) must be supplied to provide the reflux relationship. If stream 7 is calculated for example, the equipment subroutine for unit 11 calculates streams 6 and 25.





LEGEND

SAME AS FIGURE A-2

Figure C-3

Flowsheet for Solution II





Even if the balance envelopes shown on Figure C-3 were to fail, MAJOR could resort to the same trial and error calculation presented in Solution I. Stream 22 and 23 can be calculated and flag known by SEQCAL since its flow rate (zero) is specified along with any pseudo-composition (all zeros for example) and its component flow rates are unknown. For example,

|     |         |             |    |    |    |     |     |           |    |
|-----|---------|-------------|----|----|----|-----|-----|-----------|----|
| 22. | 1.      | 0.          | 0. | 0. | 0. | -1. | -1. | -1.       | 0. |
|     | overall | composition |    |    |    |     |     |           |    |
|     | mass    | (weight     |    |    |    |     |     |           |    |
|     | flow    | fraction)   |    |    |    |     |     |           |    |
|     | rate    |             |    |    |    |     |     |           |    |
|     |         |             |    |    |    |     |     | component |    |
|     |         |             |    |    |    |     |     | mass      |    |
|     |         |             |    |    |    |     |     | flows     |    |

A summary of the solutions is presented in Table C-1. The stream variable matrix is listed in Table C-2.

Note that the steam requirements for Column II vary significantly from those presented by Himmelblau. This is because he presents his physical property data as stream properties without any pressures. Thus, it was not possible to duplicate his data accurately.

Referring to Figure C-2, note that streams 1 and 2 are both specified. Had only stream 1 been specified then units 1, 2, 3, and 4 would have to be solved via the trial and error routine in PACER. This problem arises in cases such as unit 1 in Figure C-2 where the streams 2 and 3 are similar in composition and the program logic is unable to determine if the streams are intermixed. As a possible solution to this





problem, units which have no intermixing, such as heat exchangers, could have their respective process streams "paired" in the process matrix. This would be recognized by the material and energy balance routine and the compositions and flow rates of these "paired" streams equated. For example, rather than

1                    1    3   -2   -18

the process matrix row for equipment one could be

1                    (1,-2) (3,-18)



TABLE C-1

Solution to Example III

(Refer also to Table C-2)

1. Material Balance

|   | Himmelblau | Program Solution |              |
|---|------------|------------------|--------------|
|   |            | I                | II           |
| Overhead product<br>Column I (stream I)   | 1667 lb/hr | 1662.2 lb/hr     | 1667 lb/hr   |
| Reflux Column I<br>(stream 6)             | 5000 lb/hr | 4986.6 lb/hr     | 5000 lb/hr   |
| Overhead Product<br>column II (stream 16) | 1050 lb/hr | 1046.9 lb/hr     | 1052.6 lb/hr |
| Bottoms Column I<br>(stream 3)            | 8333 lb/hr | 8333 lb/hr       | 8333 lb/hr   |
| Reflux Column II<br>(stream 15)           | 3150 lb/hr | 3141 lb/hr       | 3157.9 lb/hr |
| Bottoms Column II<br>(stream 19)          | 617 lb/hr  | 612.4 lb/hr      | 614.0 lb/hr  |

2. Energy Balance

Cooling Water (lb/hr)

|                               |         |         |         |
|-------------------------------|---------|---------|---------|
| Condenser I<br>(unit 3)       | 90,000. | 84,138. | 84,363  |
| Condenser II<br>(unit 7)      | 55,200. | 34,870. | 35,060. |
| Heat Exchanger II<br>(unit 9) | 907.    | 874.    | 879.    |

Steam Loads (btu/hr)

|                              |           |           |           |
|------------------------------|-----------|-----------|-----------|
| Heat Exchanger I<br>(unit 5) | 1,125,000 | 1,051,720 | 1,054,540 |
| Column I (unit 2)            | 4,842,000 | 4,548,000 | 4,559,000 |
| Column II (unit 6)           | 1,645,000 | 700,931   | 707,618   |

Temperature (<sup>o</sup>R)

|           |       |       |       |
|-----------|-------|-------|-------|
| Stream 18 | 566.5 | 565.5 | 565.5 |
|-----------|-------|-------|-------|





TABLE C-2

(Solution II)

Stream Variables Matrix at Run 1

|         |           |            |           |               |
|---------|-----------|------------|-----------|---------------|
| 1.0000  | 1.0000    | 10000.0000 | 0.8000    | 0.1000        |
| 0.1000  | 8000.0000 | 1000.0000  | 1000.0000 | 540.0000      |
| 2.0000  | 0.0000    | 10000.0000 | 0.8000    | 0.1000        |
| 0.1000  | 8000.0000 | 1000.0000  | 1000.0000 | 630.0000      |
| 3.0000  | 0.0000    | 8333.3333  | 0.8800    | 0.0000        |
| 0.1200  | 7333.3333 | 0.0000     | 1000.0000 | 670.0000      |
| 4.0000  | 0.0000    | 6666.6666  | 0.4000    | 0.6000        |
| 0.0000  | 2666.6666 | 4000.0000  | -0.0000   | 636.0000      |
| 5.0000  | 0.0000    | 6666.6666  | 0.4000    | 0.6000        |
| 0.0000  | 2666.6666 | 4000.0000  | -0.0000   | 636.0000      |
| 6.0000  | 0.0000    | 4999.9999  | 0.4000    | 0.6000        |
| 0.0000  | 2000.0000 | 3000.0000  | -0.0000   | 636.0000      |
| 7.0000  | 0.0000    | 1666.6667  | 0.4000    | 0.6000        |
| 0.0000  | 666.6667  | 1000.0000  | -0.0000   | 636.0000      |
| 8.0000  | 2.0000    | -0.0000    | -0.0000   | -0.0000       |
| -0.0000 | -0.0000   | -0.0000    | -0.0000   | -1054538.4844 |
| 9.0000  | 2.0000    | -0.0000    | -0.0000   | -0.0000       |
| -0.0000 | -0.0000   | -0.0000    | -0.0000   | 4218153.9375  |
| 10.0000 | 0.0000    | 1666.6667  | 0.4000    | 0.6000        |
| 0.0000  | 666.6667  | 1000.0000  | -0.0000   | 636.0000      |
| 11.0000 | 0.0000    | 4210.5262  | 0.0500    | 0.9500        |
| 0.0000  | 210.5263  | 3999.9999  | 0.0000    | 632.0000      |
| 12.0000 | 2.0000    | -0.0000    | -0.0000   | -0.0000       |
| -0.0000 | -0.0000   | -0.0000    | -0.0000   | 1753002.1406  |
| 13.0000 | 2.0000    | -0.0000    | -0.0000   | -0.0000       |
| -0.0000 | -0.0000   | -0.0000    | -0.0000   | 43957.8936    |
| 14.0000 | 0.0000    | 4210.5262  | 0.0500    | 0.9500        |
| 0.0000  | 210.5263  | 3999.9999  | 0.0000    | 620.0000      |
| 15.0000 | 0.0000    | 3157.8947  | 0.0500    | 0.9500        |
| 0.0000  | 157.8947  | 2999.9999  | 0.0000    | 620.0000      |



- C11 -

|         |           |           |           |               |
|---------|-----------|-----------|-----------|---------------|
| 16.0000 | 0.0000    | 1052.6316 | 0.0500    | 0.9500        |
| 0.0000  | 52.6316   | 1000.0000 | 0.0000    | 620.0000      |
| 17.0000 | 2.0000    | 1052.6316 | 0.0500    | 0.9500        |
| 0.0000  | 52.6316   | 1000.0000 | 0.0000    | 560.0000      |
| 18.0000 | 2.0000    | 8333.3333 | 0.8800    | 0.0000        |
| 0.1200  | 7333.3333 | 0.0000    | 1000.0000 | 565.4560      |
| 19.0000 | 2.0000    | 614.0351  | 1.0000    | 0.0000        |
| 0.0000  | 614.0351  | 0.0000    | 0.0000    | 670.0000      |
| 20.0000 | 2.0000    | -0.0000   | -0.0000   | -0.0000       |
| -0.0000 | -0.0000   | -0.0000   | -0.0000   | -707618.7422  |
| 21.0000 | 2.0000    | -0.0000   | -0.0000   | -0.0000       |
| -0.0000 | -0.0000   | -0.0000   | -0.0000   | -4559567.2500 |
| 22.0000 | 1.0000    | 0.0000    | 0.0000    | 0.0000        |
| 0.0000  | 0.0000    | 0.0000    | 0.0000    | 0.0000        |
| 23.0000 | 1.0000    | 0.0000    | 0.0000    | 0.0000        |
| 0.0000  | 0.0000    | 0.0000    | 0.0000    | 0.0000        |
| 24.0000 | 0.0000    | 4999.9999 | 0.4000    | 0.6000        |
| 0.0000  | 2000.0000 | 3000.0000 | -0.0000   | 636.0000      |
| 25.0000 | 0.0000    | 1666.6667 | 0.4000    | 0.6000        |
| 0.0000  | 666.6667  | 1000.0000 | -0.0000   | 636.0000      |
| 26.0000 | 0.0000    | 3157.8947 | 0.0500    | 0.9500        |
| 0.0000  | 157.8947  | 2999.9999 | 0.0000    | 620.0000      |
| 27.0000 | 0.0000    | 1052.6316 | 0.0500    | 0.9500        |
| 0.0000  | 52.6316   | 1000.0000 | -0.0000   | 620.0000      |



APPENDIX D

Example IV

Reference: "Stoichiometry for Chemical Engineers";  
Williams and Johnson; page 258, example 6-20.

Flow Diagram: See Figure D-1

Statement: Vinylacetate monomer may be made through the  
reaction of acetylene with acetic acid:



Make-up and recycle acetic acid (HAC) is vaporized and mixed with make-up and recycle acetylene ( $\text{C}_2\text{H}_2$ ) and preheated to  $356^\circ\text{F}$ . The gaseous mixture is blown through the catalytic reactor where 60% of the HAC is converted with a 98% yield to vinylacetate (VA). The yield of VA on  $\text{C}_2\text{H}_2$  is 95%. All impurities formed are to be treated as acetaldehyde ( $\text{CH}_3\text{COH}$  or AC). The hot gaseous effluent from the reactor therefore contains  $\text{C}_2\text{H}_2$ , HAC, VA and AC. The VA effluent proceeds through a cooler to a condensation system. A small loss of  $\text{C}_2\text{H}_2$  occurs but essentially complete separation of  $\text{C}_2\text{H}_2$  from the other compounds is effected. The condensate passes to a distillation column in which AC is taken off overhead and the final product, VA, is





a side stream. The bottoms from the VA column are distilled in a second column where VA is removed overhead. A 3% loss of VA and a 4% loss of HAC can be expected in the columns. Other compositions and temperature are shown on Figure D-1.

Using this information, calculate:

- a) material balance showing amounts of each constituent in the numbered streams
- b) quantity of heat in btu/hr that must be added in the vaporizing, mixing, and preheating section
- c) size (in gallons) of the holdup tank for the Dowtherm A
- d) quantity of heat in btu/hr that must be removed in the VA condenser
- e) quantity of heat in btu/hr that must be added in the reboiler of the HAC column.

Basis:

100.0 moles of VA formed in the reactor. From this and the above information, the information in Table D-1 represents the input data calculated for PACER input.



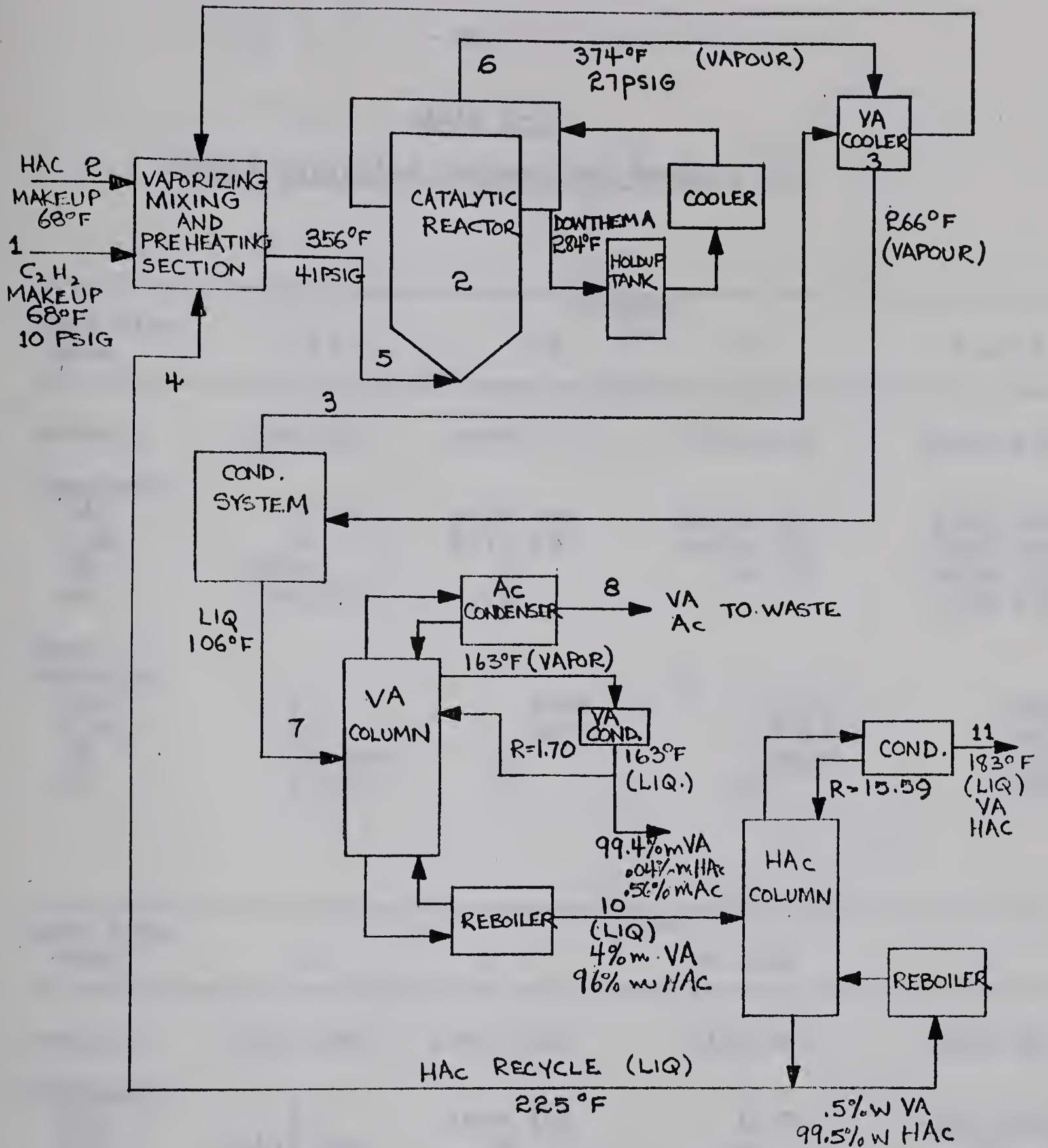


Figure D-1

Flowsheet of Vinyl Acetate Plant





TABLE D-1

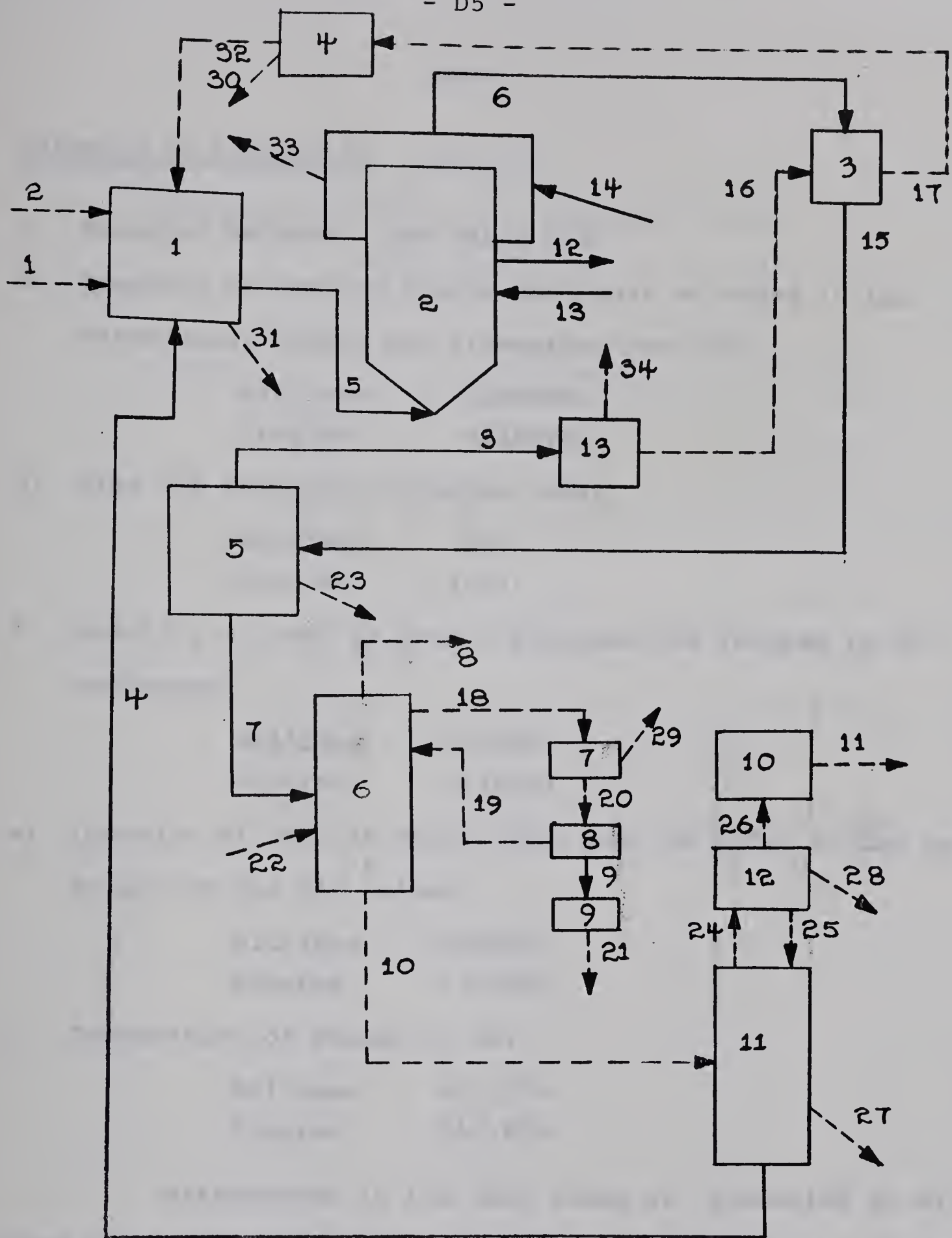
Stream Variables Vectors for Example IV

| Mass Flow<br>Rate             | Stream   |          |           |           |
|-------------------------------|----------|----------|-----------|-----------|
|                               | 13       | 12       | 5         | 6 & 15    |
| Overall                       | 8868.733 | 8868.733 | 50110.634 | 50110.635 |
| Component                     |          |          |           |           |
| HAC                           | 0.       | 6127.502 | 10216.907 | 4089.405  |
| C <sub>2</sub> H <sub>2</sub> | 0.       | 2741.231 | 39874.010 | 37132.780 |
| VA <sup>2</sup>               | 8610.    | 0.       | 19.717    | 8629.717  |
| AC                            | 258.733  | 0.       |           | 258.733   |
| Mass<br>Fraction              |          |          |           |           |
| HAC                           | 0.       | .6909    | .2039     | .08161    |
| C <sub>2</sub> H <sub>2</sub> | 0.       | .3091    | .7957     | .74102    |
| VA <sup>2</sup>               | 0.9708   | 0.       | .0039     | .17221    |
| AC                            | 0.0292   | 0.       | 0.        | .00516    |

| Mass Flow<br>Rate             | Stream     |           |          |          |
|-------------------------------|------------|-----------|----------|----------|
|                               | 3          | 7         | 9 & 21   | 4        |
| Overall                       | 37132.7801 | 12977.855 | 8376.657 | 3945.786 |
| Component                     |            |           |          |          |
| HAC                           | 0.         | 4089.405  | 1.57     | 3926.069 |
| C <sub>2</sub> H <sub>2</sub> | 37132.780  | 0.        | 0.       | 0.       |
| VA <sup>2</sup>               | 0.         | 8629.717  | 8351.109 | 19.717   |
| AC                            | 0.         | 258.733   | 24.041   | 0.       |
| Mass<br>Fraction              |            |           |          |          |
| HAC                           | 0.         | .3151     | .00028   | .995     |
| C <sub>2</sub> H <sub>2</sub> | 1.         | 0.        | 0.       | 0.       |
| VA <sup>2</sup>               | 0.         | .6650     | .997     | .005     |
| AC                            | 0.         | .0199     | .00287   | 0.       |





LEGEND

SAME AS FIGURE A-2

Figure D-2

Flowsheet for Vinyl Acetate Plant Modified  
for Computer Solution



Solution to Example IV

a) Material Balance - See Table D-2

b) Quantity of heat in btu/hr that must be added in the vaporizing, mixing and preheating sections:

|          |          |
|----------|----------|
| Williams | 7260000. |
|----------|----------|

|         |          |
|---------|----------|
| Program | 9220000. |
|---------|----------|

c) Size (in gallons) of holdup tank:

|          |      |
|----------|------|
| Williams | 800. |
|----------|------|

|         |       |
|---------|-------|
| Program | 1125. |
|---------|-------|

d) Quantity of heat in btu/hr that must be removed in VA condenser:

|          |          |
|----------|----------|
| Williams | 3660000. |
|----------|----------|

|         |          |
|---------|----------|
| Program | 4138000. |
|---------|----------|

e) Quantity of heat in btu/hr that must be added in the re-boiler of the HAC column:

|          |         |
|----------|---------|
| Williams | 985000. |
|----------|---------|

|         |          |
|---------|----------|
| Program | 1530000. |
|---------|----------|

f) Temperature of stream 17 is:

|          |                      |
|----------|----------------------|
| Williams | 655.2 <sup>°</sup> R |
|----------|----------------------|

|         |                      |
|---------|----------------------|
| Program | 656.8 <sup>°</sup> R |
|---------|----------------------|

Differences in the heat loads as presented by Williams from those obtained from the program are significant. This results again from the physical property data employed.





TABLE D-2

Solution to Example IV - Material Balance

(see Table D-3)

| Stream | Williams | Program  |
|--------|----------|----------|
| 1      | 2926.    | 2926.89  |
| 2      | 6290.    | 6291.84  |
| 8      | 269.     | 269.03   |
| 11     | 386.     | 386.38   |
| 18     | 22600.   | 22616.97 |
| 24     | 6025.    | 6025.49  |



TABLE D-3

Stream Variables Matrix at Run 4

|          |            |            |            |           |
|----------|------------|------------|------------|-----------|
| 1.0000   | 1.0000     | 2926.8940  | 0.0000     | 1.0000    |
| -0.0000  | -0.0000    | -0.0000    | 2926.8940  | -0.0000   |
| -0.0000  | 528.0000   |            |            |           |
| 2.0000   | 1.0000     | 6291.8380  | 1.0000     | -0.0000   |
| -0.0000  | -0.0000    | 6291.8380  | -0.0000    | -0.0000   |
| -0.0000  | 528.0000   |            |            |           |
| 3.0000   | 0.0000     | 37132.7798 | 0.0000     | 1.0000    |
| -0.0000  | -0.0000    | -0.0000    | 37132.7798 | -0.0000   |
| -0.0000  | 492.0000   |            |            |           |
| 4.0000   | 0.0000     | 3945.7860  | 0.9950     | 0.0000    |
| 0.0050   | 0.0000     | 3925.0690  | 0.0000     | 19.7170   |
| 0.0000   | 685.0000   |            |            |           |
| 5.0000   | 0.0000     | 50110.6338 | 0.2039     | 0.7957    |
| 0.0004   | 0.0000     | 10216.9070 | 39874.0098 | 19.7170   |
| -0.0000  | 816.0000   |            |            |           |
| 6.0000   | 0.0000     | 50110.6348 | 0.0816     | 0.7410    |
| 0.1722   | 0.0052     | 4089.4050  | 37132.7798 | 8629.7170 |
| 258.7330 | 834.0000   |            |            |           |
| 7.0000   | 0.0000     | 12977.8550 | 0.3151     | -0.0000   |
| 0.6650   | 0.0199     | 4089.4050  | 0.0000     | 8629.7170 |
| 258.7330 | 566.0000   |            |            |           |
| 8.0000   | 2.0000     | 269.0287   | -0.0000    | -0.0000   |
| 0.1290   | 0.8724     | -0.0000    | -0.0000    | 34.7069   |
| 234.6920 | 630.0000   |            |            |           |
| 9.0000   | 0.0000     | 8368.2803  | 0.0003     | 0.0000    |
| 0.9970   | 0.0029     | 2.0562     | 0.0000     | 8343.0206 |
| 24.0170  | 623.0000   |            |            |           |
| 10.0000  | 0.0000     | 4332.1693  | 0.9436     | -0.0000   |
| 0.0563   | -0.0000    | 4087.8350  | -0.0000    | 243.9011  |
| -0.0000  | 690.0000   |            |            |           |
| 11.0000  | 2.0000     | 386.3833   | 0.4213     | 0.0000    |
| 0.5802   | -0.0000    | 162.7660   | -0.0000    | 224.1841  |
| -0.0000  | 643.0000   |            |            |           |
| 12.0000  | 2.0000     | 8868.7330  | 0.6909     | 0.3091    |
| -0.0000  | -0.0000    | 6127.5020  | 2741.2310  | -0.0000   |
| -0.0000  | 10000.0000 |            |            |           |





- D9 -

|          |              |            |            |            |
|----------|--------------|------------|------------|------------|
| 13.0000  | 1.0000       | 8868.7330  | -0.0000    | -0.0000    |
| 0.9708   | 0.0292       | -0.0000    | -0.0000    | 8610.0000  |
| 258.7330 | 10000.0000   |            |            |            |
| 14.0000  | 1.0000       | -0.0000    | -0.0000    | -0.0000    |
| -0.0000  | -0.0000      | -0.0000    | -0.0000    | -0.0000    |
| -0.0000  | 5140000.0000 |            |            |            |
| 15.0000  | 0.0000       | 50110.6348 | 0.0816     | 0.7410     |
| 0.1722   | 0.0052       | 4089.4050  | 37132.7798 | 8629.7170  |
| 258.7330 | 726.0000     |            |            |            |
| 16.0000  | 0.0000       | 36947.1157 | 0.0000     | 1.0000     |
| -0.0000  | -0.0000      | -0.0000    | 36947.1157 | -0.0000    |
| -0.0000  | 492.0000     |            |            |            |
| 17.0000  | 0.0000       | 36947.1152 | 0.0000     | 1.0000     |
| -0.0000  | -0.0000      | -0.0000    | 36947.1157 | -0.0000    |
| -0.0000  | 656.8244     |            |            |            |
| 18.0000  | 0.0000       | 22616.9739 | 0.0003     | 0.0000     |
| 0.9970   | 0.0029       | 5.5573     | 0.0000     | 22548.7048 |
| 64.9107  | 623.0000     |            |            |            |
| 19.0000  | 0.0000       | 14248.6936 | 0.0003     | 0.0000     |
| 0.9970   | 0.0029       | 3.5011     | 0.0000     | 14205.6841 |
| 40.8937  | 623.0000     |            |            |            |
| 20.0000  | 0.0000       | 22616.9739 | 0.0003     | -0.0000    |
| 0.9970   | 0.0029       | 5.5573     | 0.0000     | 22548.7048 |
| 64.9107  | 623.0000     |            |            |            |
| 21.0000  | 2.0000       | 8376.6570  | 0.0003     | -0.0000    |
| 0.9970   | 0.0029       | 1.5700     | 0.0000     | 8351.1090  |
| 24.0410  | 623.0000     |            |            |            |
| 22.0000  | 1.0000       | -0.0000    | -0.0000    | -0.0000    |
| -0.0000  | -0.0000      | -0.0000    | -0.0000    | -0.0000    |
| -0.0000  | 4671683.8750 |            |            |            |
| 23.0000  | 2.0000       | -0.0000    | -0.0000    | -0.0000    |
| -0.0000  | -0.0000      | -0.0000    | -0.0000    | -0.0000    |
| -0.0000  | 8131285.1875 |            |            |            |
| 24.0000  | 0.0000       | 6410.0989  | 0.4213     | -0.0000    |
| 0.5802   | 0.0000       | 2700.2878  | 0.0000     | 3719.2147  |
| -0.0000  | 643.0000     |            |            |            |
| 25.0000  | 0.0000       | 6025.4929  | 0.4213     | 0.0000     |
| 0.5802   | 0.0000       | 2538.2705  | 0.0000     | 3496.0618  |
| -0.0000  | 643.0000     |            |            |            |



- D10 -

|         |               |            |            |          |
|---------|---------------|------------|------------|----------|
| 26.0000 | 0.0000        | 384.6059   | 0.4213     | 0.0000   |
| 0.5802  | 0.0000        | 162.0173   | 0.0000     | 223.1529 |
| -0.0000 | 643.0000      |            |            |          |
| 27.0000 | 1.0000        | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | -0.0000       | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | 1527400.0625  |            |            |          |
| 28.0000 | 2.0000        | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | -0.0000       | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | -1.0000       |            |            |          |
| 29.0000 | 2.0000        | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | -0.0000       | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | 4137978.2813  |            |            |          |
| 30.0000 | 2.0000        | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | -0.0000       | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | 168109.3711   |            |            |          |
| 31.0000 | 2.0000        | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | -0.0000       | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | -9222628.8750 |            |            |          |
| 32.0000 | 0.0000        | 36947.1152 | 0.0000     | 1.0000   |
| -0.0000 | -0.0000       | -0.0000    | 36947.1157 | -0.0000  |
| -0.0000 | 646.8244      |            |            |          |
| 33.0000 | 2.0000        | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | -0.0000       | -0.0000    | -0.0000    | -0.0000  |
| -0.0000 | 6193684.0000  |            |            |          |
| 34.0000 | 2.0000        | 185.6639   | 0.0000     | 1.0000   |
| -0.0000 | -0.0000       | -0.0000    | 185.6639   | -0.0000  |
| -0.0000 | 492.0000      |            |            |          |



APPENDIX E

Running Times for Program

| Problem     | Streams | Units | Recycle<br>Calculation<br>Used | Object Time   |
|-------------|---------|-------|--------------------------------|---------------|
| I           | 17      | 7     | No                             | 34 sec.       |
| II          | 20      | 11    | No                             | 23 sec.       |
| III         |         |       |                                |               |
| Solution I  | 21      | 9     | Yes                            | 1 min. 8 sec. |
| Solution II | 27      | 13    | No                             | 31 sec.       |
| IV          | 34      | 13    | No                             | 39 sec.       |





APPENDIX F

The following pages constitute the FORTRAN IV listings of the subroutines which make up the material and energy balance routine. None of the subroutines of the PACER executive program are listed although subroutines SCAN, MAJOR, SETUP, DREAD, GUESS1, TEST and EQUIP were altered slightly from those presented by Mosler(9). The other subroutines in the PACER executive program remain unchanged.



NOMENCLATURE

1. PACER VARIABLES

|        |  |
|--------|--|
| DELS   | stream variables test vector   |
| DELSC  | stream control variables test vector   |
| EN     | equipment parameters matrix  |
| ENC    | equipment control parameters matrix  |
| I2     | variable for PACER equipment numbers   |
| I12    | variable for PACER stream numbers  |
| KBACK  | an internal variable which is equal to zero except when the problem parameters are to be varied internally by CHANGE                               |
| KCLEAN | input indicator which controls the process of setting all matrices to zero before starting the present calculation                                 |
| KCV1   | record unknown streams assumed known in GUESS1   |
| KCV2   |  |
| KCV3   |  |
| KE1    | same list of equipments as KE3 but in order of KE2 list  |
| KE2    | first list of equipments formed by GUESS2 in attempting to use iterative solution  |
| KE3    | shortest list of equipment with which an iterative solution can be performed   |
| KEFLAG | equipment flag   |
| KES    | an indicator, $KES \neq 0$ only when an equipment was calculated directly during last scan of KPM matrix; this indicates another scan is necessary |
| KPM    | the Process Matrix   |
| KPMR   | dummy Process Matrix into which the input cards are read before storing  |





|        |   |
|--------|---|
| KPS    | the preferred streams vector  |
| KRET   | if KRET = 2 the streams assumed known in GUESS1 were not satisfactory for an iterative solution |
| KRET2  | if KRET2 = 2 calculation has failed   |
| KRET3  | indicator telling if the iterative solution converged   |
| KRUN   | run number of problem   |
| KSEM   | stream-equipment matrix   |
| KSFLAG | stream flag   |
| KSETS  | indicator controlling the amount of printout  |
| KUS    | list of unknown streams formed in GUESS1  |
| KVOID  | an indicator which will reject a problem if an equipment name has been given incorrectly        |
| LIMIT  | used in TEST to indicate if a stream has not entirely converged for all of its variables        |
| LIMIT2 | in SETUP, if LIMIT2 = 0, convergence has been obtained  |
| LIMIT3 | used in TEST to count each stream variable and stream control variable which has not converged  |
| LOOP   | counts iterative loops  |
| LOOPS  | maximum number of iterative loops allowed   |
| N2MAX  | number of rows in KPM   |
| N3MAX  | number of columns in KPM  |
| NAME   | equipment subroutine name vector  |
| NDELS  | input control variable for DELS vector  |
| NDELSC | input control variable for DELSC vector   |
| NE     | PACER equipment number  |
| NE1MAX | number of equipments in KE1 list  |



|        |  |
|--------|--|
| NE2MAX | number of equipments in KE2 list   |
| NE3MAX | number of equipments in KE3 list   |
| NECALL | vector of call numbers of equipments   |
| NECLMX | length of equipment control parameters vector  |
| NELMAX | length of equipment parameters vector  |
| NEMAX  | maximum equipment number in the problem  |
| NEQUIP | number indicating which equipment subroutine to call in calculating an equipment                 |
| NEX    | PACER equipment number of KPM row  |
| NEXEQN | external equipment number  |
| NIN    | number of input streams to a given equipment   |
| NOEN   | number of equipment parameters vectors to be read  |
| NOENC  | number of equipment control parameters vectors to be read  |
| NOGO   | input indicator determining whether a non-converged set of answers is acceptable if LOOP = LOOPS |
| NOKPM  | number of process matrix vectors to be read  |
| NOKPS  | number of preferred streams to be read   |
| NOSN   | number of stream variables vectors to be read  |
| NOSNC  | number of stream control variable vectors to be read   |
| NOUT   | number of output streams from a given equipment  |
| NPSMAX | maximum number of preferred streams to be supplied in the input                                  |
| NSCLMX | length of stream control variables vector  |
| NSLMAX | length of stream variables vector  |
| NSMAX  | maximum stream number in problem   |
| NSN    | variable for SN row number   |





|        |  |
|--------|--|
| NSNC   | variable for SNC row number  |
| NUSMAX | total number of streams listed in the unknown stream vector  |
| SN     | stream variables matrix  |
| SNC    | stream control variables matrix  |
| STRM   | dummy matrix into which input data is read before storing in proper matrix; also used for storage of information when performing an iterative solution |
| STRMC  | matrix used to store information about the SNC matrix when performing an iterative solution  |
| STRMCI | matrix used to store SNC row vectors of streams entering the equipment being calculated  |
| STRMCO | matrix used to store SNC row vectors of streams leaving the equipment being calculated   |
| STRMI  | matrix used to store SN row vectors of streams entering the equipment being calculated   |
| STRMO  | matrix used to store SN row vectors of streams leaving an equipment being calculated   |
| TITLE  | vector storing title of problem  |

## 2. MATERIAL AND ENERGY BALANCE ROUTINE VARIABLES

|        |   |
|--------|---|
| C      | vector of <u>constants</u> associated with a set of linear algebraic equations  |
| CMPROP | matrix of physical properties of the components involved. Components must be listed in the same order as in the stream variables matrix ( <u>component properties</u> matrix) |
| COEFF  | matrix of <u>coefficients</u> associated with a set of linear algebraic equations   |
| KSTR   | an indicator which counts the number of unknowns associated with a particular balance envelope. It is the length of the NSTR vector   |





|        |  |
|--------|--|
| K3STR  | stores the column number of an unknown stream variables matrix element   |
| KPHAS  | an indicator used in subroutine ENTHPY   |
| KTIMES | an indicator used in subroutines DEWPT and BUBPT   |
| LI     | a counter for input streams to a balance envelope  |
| LIP    | the length of the NIPT vector  |
| LO     | a counter for output streams from a particular balance envelope  |
| LOP    | the length of the NOPT vector  |
| LUNIT  | the length of the NUNIT vector   |
| NALLOW | a vector of the <u>numbers</u> of components which are not <u>allowed</u> to be used for a component balance   |
| NB     | an indicator. NB is greater than zero if material balance calculations around a balance were successful. It is zero otherwise.   |
| NCOMP  | the number of components in a stream (SNC(NSN,4))  |
| NEB    | an input flag. If NEB is one, only material balance calculations are to be performed. If NEB is two, both material and energy balance calculations are to be performed. If NEB is three, the same as two except without consideration of pressure in the calculation of enthalpies |
| NFLAG  | an indicator which signals subroutine SEQCAL if any calculations were made on the previous scan of the SN matrix   |
| NFLG   | an indicator used in subroutine BALNCE which signals when an attempt is being made to extend a previously located balance envelope   |
| NGAUSS | An indicator used in subroutine SETEQ which signals that there is no need to seek further equations from component balances  |
| NIPT   | a vector of numbers of streams flowing into a balance envelope   |



|        |   |
|--------|---|
| NKCOL  | a vector of column numbers of unknown elements encountered in the SN matrix when writing overall or component balance equations   |
| NKSTR  | a vector of row numbers (or stream numbers) of unknown elements in the SN matrix encountered when performing material balances  |
| NLEGTH | the length of the NSTR vector<br>(NLEGTH = LIP + LOP)   |
| NOCOMP | the number of components in the process   |
| NONOAL | the length of the NALLOW vector   |
| NOPT   | a vector of the number of streams flowing from a balance envelope   |
| NROW   | the number of equations obtained from material and energy balances around the balance envelope; also the number of rows in the COEFF matrix   |
| NSNSTR | stores the row number (stream number) of an unknown stream variables matrix element (used in SEQCAL)  |
| NSTR   | a vector of the numbers of all streams associated with a balance envelope   |
| NUBAL  | a vector of unit numbers which are to be examined after the first element of the vector has been examined; it appears in BALNCE   |
| NUBALO | a vector of unit numbers to be examined when only one element is present in the NUBAL vector; that is, when no new units were incorporated into the balance envelope on the last scan |
| NUFS   | the number of the unknown input stream which is to be solved for  |
| NUWUFS | the number of the unit into which the above stream (NUFS) flows   |
| NX9    | an indicator which rejects a certain component balance when both overall mass flow rate, mass fraction and mass component flow rate are unknown for that component (used in SETEQ)    |







|      |   |
|------|---|
| NX99 | an indicator which indicates that, if energy and material balances are to be performed simultaneously, NOCOMP component balances should be attempted in certain cases (used in SETEQ) |
| NXB1 | an indicator which signals the failure of SETEQ to effect the simultaneous calculation of the material and energy balances for a given balance envelope                               |
| TREF | the reference temperature in $^{\circ}\text{R}$   |



Index to Subroutines

| <u>Subroutine</u> | <u>Page</u> |
|-------------------|-------------|
| BALNCE            | F10         |
| SEQCAL            | F20         |
| SETEQ             | F28         |
| SIMCAL            | F39         |
| ENTHPY            | F43         |
| EQUILK            | F47         |
| QUAL              | F49         |
| BUBPT             | F52         |
| DEWPT             | F54         |



## SUBROUTINE BALNCE

```

COMMON  KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON  KEFLAG, NE, NEMAX
COMMON  KSFLAG, NS, NSMAX
COMMON  EN, NEN, NEL, NELMAX,          ENC, NENC, NECL, NE
1CLMX
COMMON  SN, NSN, NSL, NSLMAX,          SNC, NSNC, NSCL, NS
1CLMX
COMMON  KUS, NUS, NUSMAX,              KPS, NPS, NPSMAX
COMMON  NUS1, NUS2, NUS3,              KCV1, KCV2, KCV3
COMMON  KEUS, NEUS, NENUM
COMMON  KECV1, KECV2, KECV3, KECV4
COMMON  NECV1, NECV2, NECV3, NECV4
COMMON  KE1, NE1, NE1MAX,              KE2, NE2, NE2MAX
COMMON  KE3, NE3, NE3MAX,              KE4, NE4, NE4MAX
COMMON  KES, KES2, KES3
COMMON  KRET, KRET2, KRET3
COMMON  STRM, STRMC
COMMON  STRMI, STRMCI, NIN,            STRMO, STRMCO, NOUT
COMMON  DELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON  KRUN, KSETS, KCLEAN, LOOPS, NOGO
COMMON  TITLE, NOKPM, NOEN, NOENC, NOSN, NOSNC, NOKPS,
1 NSTRM
COMMON  NLX, NLY, NLXS, NLXSC
COMMON  ICONV, NEX, NECALL, NEQUIP
COMMON  ND2, ND3, ND4, ND5, ND6, ND7
COMMON  EXX, D15, D16
COMMON  I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I1
15
COMMON  J2, J3, J4, J5
COMMON  J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17, J
118, J19, J20,
1      J21, J22, J23, J24, J25, J26, J27, J28, J29, J30, J31, J3
12, J33, J34,
2      J35, J36, J37, J38, J39, J40, J41, J42, J43, J44, J45, J4
16
COMMON  J47, J48, J49, J50, J51, J52, J53, J54, J55, J56, J57, J5
18, J59, J60
COMMON  K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, K1
15, K16, K17,
1      K18, K19, K20, K21, K22, K23, K24, K25, K26, K27, K28, K2
19, K30
COMMON  L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13
COMMON  NDELS, NDELSC
COMMON  NEXEQN, NAME, KSEM
COMMON  NIPT(30), NOPT(30), LIP, LOP, NOCOMP, NUNIT(30), LUNI
1T, NB, NONOAL,
1NALLOW(15), NLEGT, NSTR(30), COEFF(16, 16), C(16), NKSTR(16
1), NKCOL(16),
2NROW, NXB1, CMPROP(15, 10), NEB, TREF
DIMENSION NEXEQN(30), NAME(30), KSEM(60, 3)
DIMENSION KPM(30, 10), KEFLAG(30), KSFLAG(60), KPMR(30
1, 12)
DIMENSION SN(60, 20), SNC(60, 10), EN(30, 10), ENC(30, 10
1)

```





```

DIMENSION KUS(60), KPS(10), KEUS(8)
DIMENSION KE1(30), KE2(30), KE3(30), KE4(30)
DIMENSION STRM(60,20), STRMC(60,10)
DIMENSION STRMI(8,20), STRMCI(8,10), STRMO(8,20), STR
1MCO(8,10)
DIMENSION DELS(20), DELSC(10)
DIMENSION TITLE(12), ICONV(15), NECALL(30)
DIMENSION NUBAL(30),NUBALO(30)

```

```

C
C SUBROUTINE BALNCE
C

```

```

C THIS SUBROUTINE ATTEMPTS TO FIND A BALANCE ENVELOPE
C WITH WHICH UNKNOWN INPUT STREAMS CAN BE CALCULATED SO
C AS TO AVOID TRIAL AND ERROR CALCULATIONS VIA PACER
C THIS SUBROUTINE IS CALLED BY MAJOR
C NUMBERS OF STREAMS ASSOCIATED WITH A PARTICULAR
C BALANCE ENVELOPE ARE STORED IN THE NIPT AND NOPT
C VECTORS
C NUMBERS OF UNITS ARE STORED IN THE NUNIT VECTOR
C

```

```

WRITE(6,1000)
1000 FORMAT(1H1,39HAN ATTEMPT WILL BE MADE TO SOLVE FOR AN
115H UNKNOWN FEED
1 /41H STREAM BY MATERIAL AND/OR ENERGY BALANCE//)
NB=0
XXU=1.1
XXL=0.99

```

```

C SEARCH FIRST FOR AN UNKNOWN FEED STREAM
C

```

```

42 CONTINUE
DO 72 I=1,30
72 NUBALO(I)=0
DO 11 LBI=1,NSMAX
IF(SN(LBI,1).LT.0.000001)GOTO11
NFLG=0

```

```

C WHEN NB IS GREATER THAN ZERO CALCULATION OF AN UNKNOWN
C INPUT STREAM WAS SUCCESSFUL
C CONTROL IS TRANSFERRED TO MAJOR
C

```

```

IF(NB.GT.0)GOTO11
IF(SN(LBI,2).GT.XXU)GOTO11
IF(SN(LBI,2).LT.XXL)GOTO11
NSIGL=0
DO 12 J=3,NSLMAX
IF(SN(LBI,J).LT.0.)NSIGL=NSIGL+1
12 CONTINUE
IF(NSIGL.LT.1)GOTO11

```

```

C NUFS IS THE NUMBER OF THE UNKNOWN STREAM
C

```

```

NUFS=LBI

```

```

C NUWUFS IS THE UNIT INTO WHICH THIS STREAM FLOWS
C

```





C

```

NUWUFS=KSEM(NUFS,3)
NFS=SN(LBI,1)
WRITE(6,100)NFS,NUWUFS

```

```

100 FORMAT(/,1X,11HUNKOWN FEED,I5,5X,7HTO UNIT,I5)

```

C

C

C

```

INITIALIZE VECTOR LENGTH COUNTERS

```

```

NUM=1

```

```

LO=1

```

```

LUNIT=0

```

```

NCHECK=0

```

```

LI=1

```

```

28 LBUN=1

```

C

C

C

C

C

C

```

THE VECTOR NUBAL CONTAINS NUMBERS OF EQUIPMENTS WHICH
ARE TO BE EXAMINED AFTER THE EQUIPMENT CURRENTLY BEING
EXAMINED (NUWUFS)
NUWUFS IS STORED IN NUBAL(1)

```

```

NUBAL(LBUN)=NUWUFS

```

```

LBUN=LBUN+1

```

C

C

C

```

THE STREAMS ASSOCIATED WITH NUWUFS ARE EXAMINED

```

```

DO 13 K3=2,N3MAX

```

```

IF(KPM(NUWUFS,K3).EQ.0)GOTO13

```

```

IF(KPM(NUWUFS,K3).GT.0)GOTO20

```

C

C

C

C

C

C

C

C

```

OUTPUT STREAMS ARE EXAMINED FOR INTER-EQUIPMENT
STREAMS WITH BITS OF INFORMATION GIVEN ABOUT THEM
AND FOR PRODUCT STREAMS
PRODUCT STREAMS ARE ENTERED DIRECTLY INTO THE BALANCE
OUTPUT STREAM VECTOR (NOPT) AS ARE TOTALLY KNOWN
INTER-EQUIPMENT STREAMS

```

```

N=-KPM(NUWUFS,K3)

```

```

IF(N.EQ.NUFS)GOTO(11,187),NUM

```

```

IF(SN(N,2).GT.1.5)GOTO19

```

```

IF(KSFLAG(N).EQ.1)GOTO19

```

```

NUN=KSEM(N,3)

```

```

NUP=LBUN-1

```

```

NCT=0

```

C

C

C

```

CHECK TO SEE IF STREAM N FLOWS BETWEEN TWO UNITS
WITHIN THE BALANCE ENVELOPE

```

C

```

IF(NUP.EQ.0)GOTO84

```

```

DO 83 ILP=1,NUP

```

```

IF(NUBAL(ILP).EQ.NUN)NCT=NCT+10

```

```

83 CONTINUE

```

```

IF(NCT.GT.0)GOTO13

```

```

84 CONTINUE

```

```

IF(LUNIT.LT.1)GOTO86

```

```

DO 85 ILP=1,LUNIT

```

WRITE (UNIT, 'CHECKING FOR CONFLICTS  
IN THE ORDER OF THE STREAMS')  
DO 100 I=1, N  
IF (UNIT .EQ. 1) WRITE (UNIT, 'CHECKING FOR CONFLICTS  
IN THE ORDER OF THE STREAMS')

IF (UNIT .EQ. 1) WRITE (UNIT, 'CHECKING FOR CONFLICTS  
IN THE ORDER OF THE STREAMS')

THE VECTOR OF THE ORDER OF THE STREAMS  
IS TO BE EXAMINED AFTER THE FIRST CONFLICT  
IS FOUND (NUMBER)  
NUMBER IS STORED IN ORDER(1)  
ORDER(1) = NUMBER  
ORDER = ORDER + 1

THE STREAM ASSOCIATED WITH ORDER IS

DO 10 I=1, N  
IF (ORDER .EQ. I) WRITE (UNIT, 'CHECKING FOR CONFLICTS  
IN THE ORDER OF THE STREAMS')

OUTPUT STREAMS ARE EXAMINED FOR INTER-  
STREAMS WITH THE INFORMATION GIVEN ABOUT THE  
AND THE OUTPUT STREAMS  
PRODUCT STREAMS ARE EXAMINED FOR INTER-  
OUTPUT STREAMS (ORDER) ARE TO BE EXAMINED  
INTER-STREAMS

IF (ORDER .EQ. I) WRITE (UNIT, 'CHECKING FOR CONFLICTS  
IN THE ORDER OF THE STREAMS')

CHECK TO SEE IF STREAM IS FIRST BETWEEN TWO  
WITHIN THE ORDER

IF (ORDER .EQ. I) WRITE (UNIT, 'CHECKING FOR CONFLICTS  
IN THE ORDER OF THE STREAMS')

IF (ORDER .EQ. I) WRITE (UNIT, 'CHECKING FOR CONFLICTS  
IN THE ORDER OF THE STREAMS')



```

      IF(NUNIT(ILP).EQ.NUN)NCT=NCT+10
85  CONTINUE
      IF(NCT.GT.0)GOTO13
86  CONTINUE
      NKPS=0
C
C      IF STREAM N IS A TOTALLY UNKNOWN STREAM OR A PREFERRED
C      STREAM THEN NUN IS TO BE ENTERED INTO THE NUBAL VECTOR
C
      IF(NOKPS.EQ.0)GOTO186
      DO 15 K=1,NOKPS
      IF(N.EQ.KPS(K)) NKPS=NKPS+10
15  CONTINUE
      IF(NKPS.GT.0)GOTO44
186  NRS=0
      DO 45 NCLMN=3,NSLMAX
      IF(SN(N,NCLMN).GT.(-.000001))NRS=NRS+10
45  CONTINUE
      IF(NRS.GT.0)GOTO19
44  NUBAL(LBUN)=NUN
      LBUN=LBUN+1
      GOTO13
19  LOP=LO-1
      NXP=0
      IF(LOP.EQ.0)GOTO91
      DO 78 IMP=1,LOP
      IF(NOPT(IMP).EQ.(-KPM(NUWUFS,K3)))NXP=NXP+1
78  CONTINUE
      IF(NXP.GT.0)GOTO13
C
C      NOPT IS A VECTOR OF OUTPUT STREAMS FROM THE BALANCE
C      ENVELOPE
C
91  NOPT(LO)=-KPM(NUWUFS,K3)
      LO=LO+1
      GOTO13
C
C      INPUT STREAMS ARE EXAMINED FOR INTER-EQUIPMENT STREAMS
C      HAVING BITS OF INFORMATION GIVEN OR FOR FEED STREAMS
C
20  N=KPM(NUWUFS,K3)
C
C      TOTALLY KNOWN STREAMS OR FEED STREAMS ARE ENTERED INTO
C      THE NIPT VECTOR
C
      IF(KSFLAG(N).EQ.1)GOTO22
      IF(SN(N,2).LT.0.99)GOTO21
22  LIP=LI-1
      NXP=0
      IF(LIP.EQ.0)GOTO90
      DO 79 IMP=1,LIP
      IF(NIPT(IMP).EQ.N)NXP=NXP+1
79  CONTINUE
      IF(NXP.GT.0)GOTO13
C

```

15. CONTINUE  
IF (A.EQ.0) GO TO 10  
DO 10 I=1,N  
IF (A.EQ.0) GO TO 10  
10 CONTINUE

[illegible]

1. [REDACTED]  
2. [REDACTED]  
3. [REDACTED]  
4. [REDACTED]  
5. [REDACTED]

1-31-60 61

00 58 18P = 1,100  
14 (18, 50, 01, 01, 01, 01)  
00 58 18P = 1,100

[illegible]

1990年10月1日  
 1990年10月1日  
 1990年10月1日

1. The information in this report is for the use of the recipient only and is not to be distributed outside the recipient's organization.

(88, 2700)  $\mu\text{m} = 0.5$

150700(1,20,(7)DAJ428)+I  
 150700(00,0,0,0,5,0)WZ)+I

[illegible]

```

C      NIPT IS A VECTOR OF NUMBERS OF STREAMS FLOWING INTO
C      THE BALANCE ENVELOPE
C
90  NIPT(LI)=N
    LI=LI+1
13  CONTINUE
    GOTO23
21  NUN=KSEM(N,2)
    NUP=LBUN-1
    NCT=0
C
C      CHECK TO SEE IF STREAM N FLOWS BETWEEN TWO UNITS
C      WITHIN THE BALANCE ENVELOPE
C
    IF(NUP.EQ.0)GOTO81
    DO 80 ILP=1,NUP
    IF(NUBAL(ILP).EQ.NUN)NCT=NCT+10
80  CONTINUE
    IF(NCT.GT.0)GOTO13
81  CONTINUE
    IF(LUNIT.LT.1)GOTO60
    DO 82 ILP=1,LUNIT
    IF(NUNIT(ILP).EQ.NUN)NCT=NCT+10
82  CONTINUE
    IF(NCT.GT.0)GOTO13
60  CONTINUE
    IF(KEFLAG(NUN).EQ.1)GOTO22
    IF(N.EQ.NUFS)GOTO22
C
C      IF STREAM N IS A TOTALLY UNKNOWN STREAM OR A PREFERRED
C      STREAM THEN UNIT NUN IS TO BE ENTERED INTO THE NUBAL
C      VECTOR
C
    IF(NCKPS.EQ.0)GOTO185
    NKPS=0
    DO 70 K=1,NOKPS
    IF(N.EQ.KPS(K))NKPS=NKPS+10
70  CONTINUE
    IF(NKPS.GT.0)GOTO44
185  NRS=0
    NUPPER=NSLMAX
    IF(NEB.EQ.1)NUPPER=NUPPER-1
    DO 56 NCLMN=3,NUPPER
    IF(SN(N,NCLMN).GT.(-0.000001))NRS=NRS+10
56  CONTINUE
    IF(NRS.GT.0)GOTO22
    GOTO44
23  CONTINUE
    IF(NCHECK.EQ.10)GOTO27
    NFP=1
    NPP=LBUN-1
    DO 26 I=1,NPP
    IF(LUNIT.EQ.0)GOTO36
    LS=0
    DO 37 J=1,LUNIT

```



11/23/1971

15(18)(1.25, 183)91

9999, 1-1 85 00

【一、二、三】=999

$$J = 1.70$$

11 (MCHSCY-60-10105)

JULI TWO

1901

15 (442.01.294)91

COMI 1105

17(28(8,8C8X))21(-1.75.10828,8)2171

10 26 0 (1994-1995)

[illegible]

電話 122-2395

$$Z = 2.88$$

18 (1982.12.29) 頁

COMET 1996B

$$IF(M, E, V, M, W, Z) = (M, W, Z) \rightarrow (M, E, V, M, W, Z)$$

29807, 1 = 35 00

0-644A

11 (NCR 92.56.0) 011111

VT 23V

550101(24JUL.02.W)91

IF (KFLAT(100).NE.1) CDLMS

JUL 11 1977

1F(ACI.GT.100)1

20011103

1. The first step is to identify the problem or question that needs to be answered. This involves understanding the context and the specific requirements of the task.

TIMOTHY, I = 9 JF 58 02

040190(1,74,71911)31

COMPLAINT

110104(0.73.108)11

CONCLUSIONS

$$f_1 + f_2 = \frac{1}{\sqrt{2}}(f_1 + f_2) = \frac{1}{\sqrt{2}}(\frac{1}{\sqrt{2}}(f_1 + f_2)) = \frac{1}{2}(f_1 + f_2)$$

012, 1 = 11 01 03

182102 (0.13.098) TL

1-12-12

1.  $\frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$

1971-72  
1972-73

(1) 1994-1995

```

      IF(NUBAL(I).EQ.NUNIT(J))LS=LS+10
37  CONTINUE
      IF(LS.GT.0)GOTO26
36  LUNIT=LUNIT+1

```

C  
C  
C  
C

NUNIT IS A VECTOR CONTAINING A LIST OF EQUIPMENT IN  
THE BALANCE ENVELOPE

```

      NUNIT(LUNIT)=NUBAL(I)
26  NUBALO(I)=NUBAL(I)
      IF(NPP.LE.1)GOTO25

```

C  
C  
C  
C

EXAMINE THE STREAMS ASSOCIATED WITH EACH UNIT IN THE  
NUBALO VECTOR

```

27  NFP=NFP+1
      IF(NFP.GT.NPP)GOTO77
      NUWUFS=NUBALO(NFP)
      NCHECK=10
      GOTO28
77  CONTINUE
      NCHECK=0
      GOTO23
25  CONTINUE
      IF(NFLG.GT.0)GOTO54

```

C  
C  
C  
C

IF NFLG IS GREATER THAN ZERO THEN AN ATTEMPT IS BEING  
MADE TO EXTEND THE BALANCE ENVELOPE

```

154 LOP=LO-1

```

C  
C  
C  
C  
C  
C  
C

CHECK TO SEE IF RECYCLE IS PRESENT WHICH RESULTS IN  
AN INTER-EQUIPMENT STREAM TO/FROM THE BALANCE ENVELOPE  
WHICH SHOULD NOW BE WITHIN THE ENVELOPE SINCE THE  
EQUIPMENT FROM/TOWHICH THIS STREAM FLOWS IS NOW WITHIN  
THE ENVELOPE

```

      LIP=LI-1
161 NELMT=0
      DO 158 NINP=1,LIP
      NSN=NIPT(NINP)
      DO 159 NU13=1,LUNIT
      IF(NUNIT(NU13).EQ.KSEM(NSN,2))NELMT=NINP
159 CONTINUE
158 CONTINUE
      IF(NELMT.EQ.0)GOTO164
      LI=LI-1
      NLIP=LI-1
      LIP=LI-1
      DO 160 NIPC=NELMT,LIP
160 NIPT(NIPC)=NIPT(NIPC+1)
      GOTO161
164 NELMTO=0
      DO 165 NOUTP=1,LOP
      NSN=NOPT(NOUTP)

```



[illegible]

```

IF (WFLD.D) GOTO 2
2: CONTINUE
GOTO 1
CHECK=0
IF CONTINUE
GOTO 2
CHECK=1
WFLD=UNVAL(177)
IF (WFLD) GOTO 1
1: CONTINUE
IF (WFLD) GOTO 1

```

1-01=98J 42

$$\begin{aligned} & 17. (M \cap I) \cap (M \cap J) = (M \cap (I \cap J)) \cap (M \cap I) \\ & \quad = (M \cap I) \cap (M \cap J) \\ & 18. (M \cap I) \cap (M \cap J) = (M \cap (I \cap J)) \cap (M \cap I) \\ & \quad = (M \cap I) \cap (M \cap J) \\ & 19. (M \cap I) \cap (M \cap J) = (M \cap (I \cap J)) \cap (M \cap I) \\ & \quad = (M \cap I) \cap (M \cap J) \end{aligned}$$

69 201103 98  
 28 201103 88  
 11 201103 77  
 11 201103 77  
 11 201103 77  
 11 201103 77  
 11 201103 77

$$\begin{aligned} W_2 &= W_1(1 + W_1) \\ G_1 &= G_2 - W_2 \\ G_2 &= G_1 + W_2 \\ W_2 &= 0 \end{aligned}$$

```

DO 166 NUI3=1,LUNIT
IF(NUNIT(NUI3).EQ.KSEM(NSN,3))NELMTO=NOUTP
166 CONTINUE
165 CONTINUE

```

```

IF(NELMTO.EQ.0)GOTO168
LO=LO-1
NLOP=LO-1
LOP=LO-1
DO 167 NOPC=NELMTO,LOP
167 NOPT(NOPC)=NOPT(NOPC+1)
GOTO164
168 WRITE(6,1001)
1001 FORMAT( 1X,37HA BALANCE ENVELOPE HAS BEEN FOUND AND
130H CALCULATION WILL BE ATTEMPTED)
WRITE(6,30)(NOPT(I),I=1,LOP)
30 FORMAT(1X,14HOUTPUT STREAMS,10I5)
WRITE(6,33)(NIPT(I),I=1,LIP)
33 FORMAT(1X,13HINPUT STREAMS,10I5)
WRITE(6,38)(NUNIT(I),I=1,LUNIT)
38 FORMAT(1X,29HUNITS IN BALANCE ENVELOPE ARE,10I5)
NXB1=0

```

C

C

C

ATTEMPT THE CALCULATION OF THE BALANCE ENVELOPE

CALL SEQCAL

C

C

C

IF NB IS GREATER THAN ZERO CALCULATIONS HAVE BEEN  
SUCCESSFUL

C

```

IF(NB.GT.0)GOTO11
IF(NFLG.GT.0)GOTO50
IF(NB.EQ.0)GOTO50
11 CONTINUE
IF(NB.GT.0)GOTO40
IF(XXU.LT.0.5)GOTO40

```

C

C

C

C

NO FEED STREAMS HAVE BEEN SUCCESSFULLY CALCULATED SO  
SEARCH FOR AN UNKNOWN INTER-EQUIPMENT STREAM

XXU=0.1

XXL=-0.1

GOTO42

50 CONTINUE

C

C

C

C

C

EXAMINE THE OUTPUT STREAMS FOR INTER-EQUIPMENT STREAMS  
THEN EXTEND THE ENVELOPE USING THESE STREAMS IN SEARCH  
OF A NEW ENVELOPE

NUM=2

NLOP=LOP

NLOP1=NLOP

NLIP=LIP

NLOP2=NLOP

NLIP2=NLIP

NLUNIT=LUNIT





```

DO 51 N2=1,NLOP1
NSN=NOPT(N2)
IF(SN(NSN,2).LT.0.5)GOTO52
51 CONTINUE
GOTO11
40 CONTINUE
RETURN
52 CONTINUE
NSUM=0

C
C DO NOT EXTEND TOTALLY KNOWN INTER-EQUIPMENT STREAMS
C
IF(KSFLAG(NSN).GT.0)GOTO51
NCP3=NOCOMP+3
NSCMX=0

C
C NSUM IS THE NUMBER OF UNKNOWNNS ASSOCIATED WITH THE
C ENVELOPE TO BE EXTENDED
C
DO 66 I=1,NLOP
NSN=NOPT(I)
NSC=SNC(NSN,4)+0.5
IF(NSC.GT.NSCMX)NSCMX=NSC
DO 53 K3=3,NCP3
IF(SN(NSN,K3).LT.0.)NSUM=NSUM+1
53 CONTINUE
KK3=2*NOCOMP+4
IF(SN(NSN,KK3).LT.0.)NSUM=NSUM+1
66 CONTINUE
DO 67 I=1,NLIP
NSN=NIPT(I)
NSC=SNC(NSN,4)+0.5
IF(NSC.GT.NSCMX)NSCMX=NSC
DO 68 K3=3,NCP3
IF(SN(NSN,K3).LT.0.)NSUM=NSUM+1
68 CONTINUE
IF(SN(NSN,KK3).LT.0.)NSUM=NSUM+1
67 CONTINUE
NSN=NOPT(N2)
NUWUFS=KSEM(NSN,3)
NCHECK=0
NFLG=NFLG+1
GOTO28
54 LOP=LO-1
NSUM1=0
NSCMN=0
DO 59 N3= 1,LOP
NSN=NOPT(N3)
NCHT=0
DO 71 J=1,LUNIT
IF(KSEM(NSN,3).EQ.NUNIT(J))NCHT=NCHT+1
71 CONTINUE
IF(NCHT.GT.0)GOTO59
NSC=SNC(NSN,4)+0.5
IF(NSC.GT.NSCMN)NSCMN=NSC

```

$11(2k)(2k+1)(2k+2)(2k+3)(2k+4)(2k+5)(2k+6)(2k+7)(2k+8)(2k+9)(2k+10)(2k+11)(2k+12)(2k+13)(2k+14)(2k+15)(2k+16)(2k+17)(2k+18)(2k+19)(2k+20)(2k+21)(2k+22)(2k+23)(2k+24)(2k+25)(2k+26)(2k+27)(2k+28)(2k+29)(2k+30)(2k+31)(2k+32)(2k+33)(2k+34)(2k+35)(2k+36)(2k+37)(2k+38)(2k+39)(2k+40)(2k+41)(2k+42)(2k+43)(2k+44)(2k+45)(2k+46)(2k+47)(2k+48)(2k+49)(2k+50)(2k+51)(2k+52)(2k+53)(2k+54)(2k+55)(2k+56)(2k+57)(2k+58)(2k+59)(2k+60)(2k+61)(2k+62)(2k+63)(2k+64)(2k+65)(2k+66)(2k+67)(2k+68)(2k+69)(2k+70)(2k+71)(2k+72)(2k+73)(2k+74)(2k+75)(2k+76)(2k+77)(2k+78)(2k+79)(2k+80)(2k+81)(2k+82)(2k+83)(2k+84)(2k+85)(2k+86)(2k+87)(2k+88)(2k+89)(2k+90)(2k+91)(2k+92)(2k+93)(2k+94)(2k+95)(2k+96)(2k+97)(2k+98)(2k+99)(2k+100)$

```
0Y CONTINUE  
IF(SUB(K,K),L,L)/N=SUB+1  
86 CONTINUE
```

$$\begin{aligned} (SW) T_{90W} &= \mu_{2A} \\ (E, \mu_{2A}) \mu_{2A} &= \mu_{2A}(WUW) \\ &= \mu_{2A}H\mu_{2A} \\ [+2] T_{90W} &= [-2] T_{90W} \\ &= \mu_{2A}T_{90W} \\ [-1] T_{90W} &= \mu_{2A}T_{90W} \end{aligned}$$

$W2001 = 0$   
 $W2004 = 0$   
 $W2007 = 0$   
 $W2010 = 0$   
 $W2013 = 0$   
 $W2016 = 0$   
 $W2019 = 0$   
 $W2022 = 0$   
 $W2025 = 0$   
 $W2028 = 0$   
 $W2031 = 0$   
 $W2034 = 0$   
 $W2037 = 0$   
 $W2040 = 0$   
 $W2043 = 0$   
 $W2046 = 0$   
 $W2049 = 0$   
 $W2052 = 0$   
 $W2055 = 0$   
 $W2058 = 0$   
 $W2061 = 0$   
 $W2064 = 0$   
 $W2067 = 0$   
 $W2070 = 0$   
 $W2073 = 0$   
 $W2076 = 0$   
 $W2079 = 0$   
 $W2082 = 0$   
 $W2085 = 0$   
 $W2088 = 0$   
 $W2091 = 0$   
 $W2094 = 0$   
 $W2097 = 0$   
 $W2100 = 0$   
 $W2103 = 0$   
 $W2106 = 0$   
 $W2109 = 0$   
 $W2112 = 0$   
 $W2115 = 0$   
 $W2118 = 0$   
 $W2121 = 0$   
 $W2124 = 0$   
 $W2127 = 0$   
 $W2130 = 0$   
 $W2133 = 0$   
 $W2136 = 0$   
 $W2139 = 0$   
 $W2142 = 0$   
 $W2145 = 0$   
 $W2148 = 0$   
 $W2151 = 0$   
 $W2154 = 0$   
 $W2157 = 0$   
 $W2160 = 0$   
 $W2163 = 0$   
 $W2166 = 0$   
 $W2169 = 0$   
 $W2172 = 0$   
 $W2175 = 0$   
 $W2178 = 0$   
 $W2181 = 0$   
 $W2184 = 0$   
 $W2187 = 0$   
 $W2190 = 0$   
 $W2193 = 0$   
 $W2196 = 0$   
 $W2199 = 0$   
 $W2202 = 0$   
 $W2205 = 0$   
 $W2208 = 0$   
 $W2211 = 0$   
 $W2214 = 0$   
 $W2217 = 0$   
 $W2220 = 0$   
 $W2223 = 0$   
 $W2226 = 0$   
 $W2229 = 0$   
 $W2232 = 0$   
 $W2235 = 0$   
 $W2238 = 0$   
 $W2241 = 0$   
 $W2244 = 0$   
 $W2247 = 0$   
 $W2250 = 0$   
 $W2253 = 0$   
 $W2256 = 0$   
 $W2259 = 0$   
 $W2262 = 0$   
 $W2265 = 0$   
 $W2268 = 0$   
 $W2271 = 0$   
 $W2274 = 0$   
 $W2277 = 0$   
 $W2280 = 0$   
 $W2283 = 0$   
 $W2286 = 0$   
 $W2289 = 0$   
 $W2292 = 0$   
 $W2295 = 0$   
 $W2298 = 0$   
 $W2301 = 0$   
 $W2304 = 0$   
 $W2307 = 0$   
 $W2310 = 0$   
 $W2313 = 0$   
 $W2316 = 0$   
 $W2319 = 0$   
 $W2322 = 0$   
 $W2325 = 0$   
 $W2328 = 0$   
 $W2331 = 0$   
 $W2334 = 0$   
 $W2337 = 0$   
 $W2340 = 0$   
 $W2343 = 0$   
 $W2346 = 0$   
 $W2349 = 0$   
 $W2352 = 0$   
 $W2355 = 0$   
 $W2358 = 0$   
 $W2361 = 0$   
 $W2364 = 0$   
 $W2367 = 0$   
 $W2370 = 0$   
 $W2373 = 0$   
 $W2376 = 0$   
 $W2379 = 0$   
 $W2382 = 0$   
 $W2385 = 0$   
 $W2388 = 0$   
 $W2391 = 0$   
 $W2394 = 0$   
 $W2397 = 0$   
 $W2400 = 0$   
 $W2403 = 0$   
 $W2406 = 0$   
 $W2409 = 0$   
 $W2412 = 0$   
 $W2415 = 0$   
 $W2418 = 0$   
 $W2421 = 0$   
 $W2424 = 0$   
 $W2427 = 0$   
 $W2430 = 0$   
 $W2433 = 0$   
 $W2436 = 0$   
 $W2439 = 0$   
 $W2442 = 0$   
 $W2445 = 0$   
 $W2448 = 0$   
 $W2451 = 0$   
 $W2454 = 0$   
 $W2457 = 0$   
 $W2460 = 0$   
 $W2463 = 0$   
 $W2466 = 0$   
 $W2469 = 0$   
 $W2472 = 0$   
 $W2475 = 0$   
 $W2478 = 0$   
 $W2481 = 0$   
 $W2484 = 0$   
 $W2487 = 0$   
 $W2490 = 0$   
 $W2493 = 0$   
 $W2496 = 0$   
 $W2499 = 0$   
 $W2502 = 0$   
 $W2505 = 0$   
 $W2508 = 0$   
 $W2511 = 0$   
 $W2514 = 0$   
 $W2517 = 0$   
 $W2520 = 0$   
 $W2523 = 0$   
 $W2526 = 0$   
 $W2529 = 0$   
 $W2532 = 0$   
 $W2535 = 0$   
 $W2538 = 0$   
 $W2541 = 0$   
 $W2544 = 0$   
 $W2547 = 0$   
 $W2550 = 0$   
 $W2553 = 0$   
 $W2556 = 0$   
 $W2559 = 0$   
 $W2562 = 0$   
 $W2565 = 0$   
 $W2568 = 0$   
 $W2571 = 0$   
 $W2574 = 0$   
 $W2577 = 0$   
 $W2580 = 0$   
 $W2583 = 0$   
 $W2586 = 0$   
 $W2589 = 0$   
 $W2592 = 0$   
 $W2595 = 0$   
 $W2598 = 0$   
 $W2601 = 0$   
 $W2604 = 0$   
 $W2607 = 0$   
 $W2610 = 0$   
 $W2613 = 0$   
 $W2616 = 0$   
 $W2619 = 0$   
 $W2622 = 0$   
 $W2625 = 0$   
 $W2628 = 0$   
 $W2631 = 0$   
 $W2634 = 0$   
 $W2637 = 0$   
 $W2640 = 0$   
 $W2643 = 0$   
 $W2646 = 0$   
 $W2649 = 0$   
 $W2652 = 0$   
 $W2655 = 0$   
 $W2658 = 0$   
 $W2661 = 0$   
 $W2664 = 0$   
 $W2667 = 0$   
 $W2670 = 0$   
 $W2673 = 0$   
 $W2676 = 0$   
 $W2679 = 0$   
 $W2682 = 0$   
 $W2685 = 0$   
 $W2688 = 0$   
 $W2691 = 0$   
 $W2694 = 0$   
 $W2697 = 0$   
 $W2700 = 0$   
 $W2703 = 0$   
 $W2706 = 0$   
 $W2709 = 0$   
 $W2712 = 0$   
 $W2715 = 0$   
 $W2718 = 0$   
 $W2721 = 0$

[illegible]



LIMIT=NOCOMP+3

C  
C  
C  
C

NSUM1 IS THE NUMBER OF UNKNOWNNS ASSOCIATED WITH THE  
EXTENDED ENVELOPE

```

DO 55 K3=3,LIMIT
  IF(SN(NSN,K3).LT.0.)NSUM1=NSUM1+1
55 CONTINUE
  IF(SN(NSN,KK3).LT.0.)NSUM1=NSUM1+1
59 CONTINUE
  LIP=LI-1
  DO 65 N3=      1,LIP
    NSN=NIPT(N3)
    NCHT=0
    DO 69 J=1,LUNIT
      IF(KSEM(NSN,2).EQ.NUNIT(J))NCHT=NCHT+1
69 CONTINUE
      IF(NCHT.GT.0)GOTO65
      NSC=SNC(NSN,4)+0.5
      IF(NSC.GT.NSCMN)NSCMN=NSC
      DO 61 K3=3,LIMIT
        IF(SN(NSN,K3).LT.0.)NSUM1=NSUM1+1
61 CONTINUE
        IF(SN(NSN,KK3).LT.0.)NSUM1=NSUM1+1
65 CONTINUE
        IF(NSUM1.LE.NSUM)GOTO57
        IF(NSCMN.GT.NSCMX)GOTO57
        NLOW1=NLIP2+1
        LBUN=1
        IF(NLOW1.GT.LIP)GOTO64

```

C  
C  
C  
C  
C

EXTEND THIS ENVELOPE FURTHER ALONG INTER-EQUIPMENT  
STREAMS IF IT IS LESS ATTRACTIVE THAN THE INITIAL  
BALANCE ENVELOPE

```

DO 62 N3=NLOW1,LIP
  IF(LBUN.GT.1)GOTO62
  NSN=NIPT(N3)
  IF(SN(NSN,2).GT.0.5)GOTO62
  NUWUFS=KSEM(NSN,2)
  NLNN3=0
  DO 725 NN3=3,NSLMAX
    IF(SN(NSN,NN3).LT.0.)NLNN3=NLNN3+1
725 CONTINUE
    IF(NLNN3.EQ.0)GOTO62
    LBUN=LBUN+1
62 CONTINUE
    NLIP2=LI-1
    NCHECK=0
    IF(LBUN.GT.1)GOTO28
64 NLOW1=NLOP2+1
    IF(NLOW1.GT.LOP)GOTO187
    DO 63 N3=NLOW1,LOP
      IF(LBUN.GT.1)GOTO63
      NSN=NOPT(N3)

```



```

      IF(SN(NSN,2).GT.0.5)GOTO63
      NUWUFS=KSEM(NSN,3)
      NLNN3=0
      DO 726 NN3=3,NSLMAX
      IF(SN(NSN,NN3).LT.0.)NLNN3=NLNN3+1
726  CONTINUE
      IF(NLNN3.EQ.0)GOTO63
      LBUN=LBUN+1
63  CONTINUE
      NLOP2=LO-1
      NCHECK=0
      IF(LBUN.GT.1)GOTO28
C
C      RETURN ENVELOPE TO INCLUDE THE ORIGINAL UNITS AND
C      SEARCH FOR ANOTHER INTER-EQUIPMENT OUTPUT STREAM ALONG
C      WHICH TO EXTEND THE ENVELOPE
C
187  LO=NLOP+1
      LI=NLIP+1
      LUNIT=NLUNIT
      NLIP2=NLIP
      NLOP2=NLOP
      GOTO51
C
C      REMOVE THE INTER-EQUIPMENT OUTPUT STREAM ALONG WHICH
C      THE INITIAL BALANCE ENVELOPE WAS EXTENDED FROM THE
C      NOPT VECTOR IF A MORE ATTRACTIVE ENVELOPE WAS LOCATED
C
57  NOPT(N2)=NOPT(NLOP+1)
      NLOW=NLOP+2
      NLOP=LO-1
      DO 58 I=NLOW,NLOP
58  NOPT(I-1)=NOPT(I)
      NLIP=LI-1
      NLUNIT=LUNIT
      LO=LO-1
      NLOP=LO-1
      GOTO154
      END

```





```

SUBROUTINE SEQCAL
COMMON  KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON  KEFLAG, NE, NEMAX
COMMON  KSFLAG, NS, NSMAX
COMMON  EN, NEN, NEL, NELMAX,          ENC, NENC, NECL, NE
1CLMX
COMMON  SN, NSN, NSL, NSLMAX,          SNC, NSNC, NSCL, NS
1CLMX
COMMON  KUS, NUS, NUSMAX,              KPS, NPS, NPSMAX
COMMON  NUS1, NUS2, NUS3,              KCV1, KCV2, KCV3
COMMON  KEUS, NEUS,  NENUM
COMMON  KECV1, KECV2, KECV3, KECV4
COMMON  NECV1, NECV2, NECV3, NECV4
COMMON  KE1, NE1, NE1MAX,              KE2, NE2, NE2MAX
COMMON  KE3, NE3, NE3MAX,              KE4, NE4, NE4MAX
COMMON  KES, KES2, KES3
COMMON  KRET, KRET2, KRET3
COMMON  STRM, STRMC
COMMON  STRMI, STRMCI, NIN,            STRMO, STRMCO, NOUT
COMMON  DELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON  KRUN, KSETS, KCLEAN, LOOPS, NOGO
COMMON  TITLE, NOKPM, NOEN, NOENC, NOSN, NOSNC, NOKPS,
1 NSTRM
COMMON  NLX, NLY, NLXS, NLXSC
COMMON  ICONV, NEX, NECALL, NEQUIP
COMMON  ND2, ND3, ND4, ND5, ND6, ND7
COMMON  EXX, D15, D16
COMMON  I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I1
15
COMMON  J2, J3, J4, J5
COMMON  J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17, J
118, J19, J20,
1      J21, J22, J23, J24, J25, J26, J27, J28, J29, J30, J31, J3
12, J33, J34,
2      J35, J36, J37, J38, J39, J40, J41, J42, J43, J44, J45, J4
16
COMMON  J47, J48, J49, J50, J51, J52, J53, J54, J55, J56, J57, J5
18, J59, J60
COMMON  K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, K1
15, K16, K17,
1      K18, K19, K20, K21, K22, K23, K24, K25, K26, K27, K28, K2
19, K30
COMMON  L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13
COMMON  NDELS, NDELSC
COMMON  NEXEQN, NAME, KSEM
COMMON  NIPT(30), NOPT(30), LIP, LOP, NOCCMP, NUNIT(30), LUNI
1T, NB, NONOAL,
INALLCW(15), NLEGTH, NSTR(30), COEFF(16,16), C(16), NKSTR(16
1), NKCOL(16),
2NROW, NXB1, CMPROP(15,10), NEB, TREF
DIMENSION NEXEQN(30), NAME(30), KSEM(60,3)
DIMENSION  KPM(30,10), KEFLAG(30), KSFLAG(60), KPMR(30
1,12)
DIMENSION  SN(60,20), SNC(60,10), EN(30,10), ENC(30,10
1)

```





```

DIMENSION KUS(60), KPS(10), KEUS(8)
DIMENSION KE1(30), KE2(30), KE3(30), KE4(30)
DIMENSION STRM(60,20), STRMC(60,10)
DIMENSION STRMI(8,20), STRMCI(8,10), STRMO(8,20), STR
IMCO(8,10)
DIMENSION DELS(20), DELSC(10)
DIMENSION TITLE(12), ICONV(15), NECALL(30)

```

C

C

SUBROUTINE SEQCAL

C

C

THIS SUBROUTINE ATTEMPTS SEQUENTIAL CALCULATIONS

C

THAT IS THE SOLUTION OF ONE EQUATION IN ONE UNKNOWN

C

C

EQUATIONS USED ARE

C

C

I SUMMATION OVER NSN OF  $S(NSN)*SN(NSN,3)=0$ .

C

II SUMMATION OVER NSN OF  $S(NSN)*SN(NSN,J+3)=0$ .

C

FOR  $J=1,2,...,NOCOMP$ 

C

III SUMMATION OVER J OF  $SN(NSN,J+3)=SN(NSN,3)$ 

C

IV  $SN(NSN,J+3)=SN(NSN,J+NOCOMP+3)/SN(NSN,3)$ 

C

NLENGH=LIP+LOP

DO 10 I=1,LIP

10 NSTR(I)=NIPT(I)

DO 11 I=1,LOP

NSUB=LIP+I

11 NSTR(NSUB)=NOPT(I)

40 NBIG=3

NEND=3

NFLAG=0

20 CONTINUE

C

C

USING EQUATIONS I AND II ATTEMPT TO SOLVE FOR AN

C

OVERALL MASS FLOW RATE OR A COMPONENT FLOW RATE

C

DO 12 K3=NBIG,NEND

NFG=0

IF(NCNOAL.EQ.0)GOTO201

NCT=0

DO 200 NP=1,NONCAL

IF(NALLOW(NP).EQ.K3-6)NCT=NCT-1

200 CONTINUE

IF(NCT.LT.0)GOTO12

201 CONTINUE

DO 13 I=1,NLENGH

NSN=NSTR(I)

IF(NEND.GT.3)GOTO1114

NSCRPT=2\*NOCOMP+4

IF(ABS(SN(NSN,NSCRPT)-10000.).LT.0.001)GOTO13

1114 IF(SN(NSN,K3).LT.0.)GOTO14

GOTO13

14 NFG=NFG+1

NSNSTR=NSN

13 CONTINUE

IF(NFG.EQ.1)GOTO15





```

      GOTO12
15  SIN=C.
      DO 16 I=1,LIP
      IF(NIPT(I).EQ.NSNSTR)GOTO16
      NSN=NIPT(I)
      SIN=SIN+SN(NSN,K3)
16  CONTINUE
      SOUT=0.
      DO 17 I=1,LOP
      IF(NOPT(I).EQ.NSNSTR)GOTO17
      NSN=NOPT(I)
      SOUT=SOUT+SN(NSN,K3)
17  CONTINUE
      NFLAG=NFLAG+1
      IF(SN(NSNSTR,2).GT.1.1)GOTO18
      IF(SN(NSNSTR,2).LT.0.99)GOTO118
120  SN(NSNSTR,K3)=SOUT-SIN
      IF(SN(NSNSTR,K3).LT.0.)SN(NSNSTR,K3)=0.
      GOTO12
118  CONTINUE
      NX3=0
      DO 119 N5=1,LIP
      IF(NSNSTR.EQ.NIPT(N5))NX3=NX3+2
119  CONTINUE
      IF(NX3.GT.0)GOTO120
18  SN(NSNSTR,K3)=SIN-SOUT
      IF(SN(NSNSTR,K3).LT.0.)SN(NSNSTR,K3)=0.
12  CONTINUE
      IF(NBIG.GT.3)GOTO19
      NBIG=NOCOMP+4
      NEND=2*NOCOMP+3
      GOTO20
19  CONTINUE
C
C   USING EQUATION III SCAN THE SN MATRIX IN AN ATTEMPT
C   TO CALCULATE AN UNKNOWN ELEMENT
C
      DO 21 I=1,NLEGT
      NSN=NSTR(I)
      NBIG=3
      NEND=3
      NFG=0
      IF(KSFLAG(NSN).EQ.1)GOTO21
      IF(KSFLAG(NSN).EQ.7)GOTO21
25  CONTINUE
      DO 1122 K3=NBIG,NEND
      IF(SN(NSN,K3).LT.0.)GOTO23
      GOTO1122
23  NFG=NFG+1
      K3STR=K3
1122 CONTINUE
      IF(NBIG.GT.3)GOTO24
      NBIG=NOCOMP+4
      NEND=2*NOCOMP+3
      GOTO25

```





```

24 CONTINUE
   IF(NFG.EQ.1)GOTO26
   GOTO21
26 SCOMP=0.
   DO 27 K3=NBIG,NEND
   IF(K3.EQ.K3STR)GOTO27
   SCOMP=SCOMP+SN(NSN,K3)
27 CONTINUE
   NFLAG=NFLAG+1
   IF(K3STR.EQ.3)GOTO28
   SN(NSN,K3STR)=SN(NSN,3)-SCOMP
   IF(SN(NSN,K3STR).LT.0.)SN(NSN,K3STR)=0.
   GOTO21
28 SN(NSN,3)=SCOMP
   IF(SN(NSN,3).LT.0.)SN(NSN,3)=0.
21 CONTINUE
C
C   USING EQUATION IV ATTEMPT TO SOLVE FOR AN UNKNOWN
C   ELEMENT OF THE SN MATRIX
C
   DO 29 I=1,NLENGTH
   NSN=NSTR(I)
   NUL=NOCOMP+3
   DO 32 K3=4,NUL
   NFG=0
   IF(SN(NSN,3).LT.0.)GOTO30
   GOTO31
30 NFG=NFG+1
   K3STR=3
31 CONTINUE
   IF(SN(NSN,K3).LT.0.)GOTO34
   GOTO33
34 NFG=NFG+1
   K3STR=K3
33 CONTINUE
   NN=K3+NOCOMP
   IF(SN(NSN,NN).LT.0.)GOTO36
   GOTO35
36 NFG=NFG+1
   K3STR=K3+NCCOMP
35 CONTINUE
   IF(NFG.EQ.1)GOTO37
   GOTO32
37 NFLAG=NFLAG+1
   IF(K3STR.EQ.3)GOTO38
   IF(K3STR.LE.NOCOMP+3)GOTO39
   NU=K3STR-NOCOMP
   SN(NSN,K3STR)=SN(NSN,3)*SN(NSN,NU)
   GOTO32
38 NU=K3+NOCOMP
   IF(SN(NSN,K3).LT.0.01)GOTO22
   SN(NSN,3)=SN(NSN,NU)/SN(NSN,K3)
   GOTO32
22 NFLAG=NFLAG-1
   GOTO32

```



```

39 NU=K3STR+NOCOMP
   SN(NSN,K3STR)=SN(NSN,NU)/SN(NSN,3)
32 CONTINUE
29 CONTINUE
C
C   IF ANY CALCULATIONS WERE MADE THEN REPEAT THE CYCLE
C   THAT IS IF NFLAG IS GREATER THAN ZERO
C
C   IF(NFLAG.GT.0)GOTO40
C
C   CHECK TO SEE IF A SIMULTANEOUS SOLUTION IS REQUIRED
C
C   NC1=0
C   DO 41 I=1,NLENGH
C   NC=0
C   NSN=NSTR(I)
C   NLT=NOCOMP+3
C   DO 441 K3=4,NLT
C   IF(SN(NSN,K3).LT.0.)NC=NC+1
441 CONTINUE
C   IF(NC.GT.0)NC1=NC1+1
C   IF(NC.GT.0)GOTO41
C
C   CHECK TO SEE IF IT IS POSSIBLE TO CALCULATE ANY
C   QUALITIES, DEW POINTS, OR BUBBLE POINTS
C
C   IF(NEB.EQ.1)GOTO419
C   K3=2*NOCOMP+4
C   N=NSN
C   NSCRPT=2*NOCOMP+4
C   IF(ABS(SN(N,NSCRPT)-10000.).LT.0.001)GOTO419
C   IF(SNC(N,3).LT.C.)GOTO419
C   IF(SNC(N,6).LT.0.)CALLBUBPT(N)
C   IF(SNC(N,7).LT.0.)CALL DEWPT(N)
C   IF(SN(N,K3).LT.0.)GOTO419
C   IF(SNC(N,5).LT.0.)CALL QUAL(N)
419 CONTINUE
C   IF(SN(NSN,3).LT.0.)NC1=NC1+1
41 CONTINUE
C   IF(NC1.GT.0)GOTO42
C   GOTO43
42 WRITE(6,44)
C
C   SIMULTANEOUS SOLUTION REQUIRED
C
44 FORMAT(1X,30HSIMULTANEOUS SOLUTION REQUIRED)
C   CALL SETEQ
C
C   IF NXB1 IS -1 THEN A SIMULTANEOUS SOLUTION IS NOT
C   POSSIBLE AND CONTROL IS RETURNED TO BALNCE
C
C   IF(NXB1.LT.0)GOTO300
C
C   IF NXB1 IS 16 THEN THE EQUIPMENT BEING CALCULATED
C   CANNOT BE SOLVED BY MATERIAL AND/OR ENERGY BALANCE

```





```

C      AND CONTROL IS RETURNED TO SETUP OR EQUIP
C
      IF(NXB1.EQ.16)RETURN
      GOTO40
43 NCEB=0
C
C      IF ONLY MATERIAL BALANCE IS REQUIRED THEN GOTO THE
C      NEXT SECTION
C
      IF(NEB.EQ.1)GOTO300
C
C      ENERGY BALANCE CALCULATION
C
      DO 45 I=1,NLEGT
      NSN=NSTR(I)
      IF(SNC(NSN,3).LT.0.)NCEB=10
      NSBCRT=2*NOCOMP+4
      NZER=0
      NUPPER=NOCOMP+3
      DO 1145 IHT=4,NUPPER
      IF(ABS(SN(NSN,IHT)).LT.0.0001)NZER=NZER+1
1145 CONTINUE
      IF(NZER.EQ.NOCOMP)GOTO1146
      IF(SN(NSN,NSBCRT).LT.0.)GOTO46
      GOTO45
1146 IF(ABS(SN(NSN,NSBCRT)+1.).LT.0.0001)GOTO46
      GOTO45
46 NCEB=NCEB+1
      NSNSTR=NSN
45 CONTINUE
C
C      IF ONLY ONE TEMPERATURE OR HEAT FLOW IS UNKNOWN THEN
C      ATTEMPT A SEQUENTIAL ENERGY BALANCE SOLUTION SINCE
C      ALL MASS FLOW DATA IS KNOWN
C
      IF(NCEB.EQ.1)GOTO47
      IF(NCEB.EQ.0)GOTO300
      WRITE(6,54)
54 FORMAT(1X,30HUNABLE TO SOLVE ENERGY BALANCE)
      NB=NB+1
      GOTO300
47 HEAT=0.
      DO 48 I=1,NLEGT
      NSN=NSTR(I)
      IF(NSN.EQ.NSNSTR)GOTO48
      NX3=C
      DO 122 N5=1,LIP
      IF(NSN.EQ.NIPT(N5))NX3=NX3+2
122 CONTINUE
      SUM2=0.
      IF(ABS(SN(NSN,NSBCRT)-10000.).LT.0.001)GOTO48
      NUPE=NOCOMP+3
      NZER=0
      DO 124 NP=4,NUPE
      K3=NSBCRT

```





```

      IF (ABS(SN(NSN,NP)).LT.0.0001)GOTO418
      CALL ENTHPY(ANS,NP-3)
      SUM2=SUM2+SN(NSN,NP)*ANS
      GOTO124
418  NZER=NZER+1
124  CONTINUE
      IF(NZER.EQ.NOCOMP)GOTO416
      IF(NX3.GT.0)GOTO123
      HEAT=HEAT+SUM2*SN(NSN,3)
      GOTO48
416  NHE=2*NOCOMP+4
      IF(NX3.GT.0)GOTO1416
      HEAT=HEAT+SN(NSN,NHE)
      GOTO48
1416 HEAT=HEAT-SN(NSN,NHE)
      GOTO48
123  HEAT=HEAT-SUM2*SN(NSN,3)
      48  CONTINUE
      NX3=0
      DO 430 I=1,LIP
      IF(NSNSTR.EQ.NIPT(I))NX3=NX3+1
430  CONTINUE
      IF(NX3.EQ.0)HEAT=-HEAT
C
C      SOLVE FOR UNKNOWN HEAT FLOW OR TEMPERATURE
C      AN ITERATIVE SCHEME IS REQUIRED TO SOLVE FOR AN
C      UNKNOWN TEMPERATURE
C
      NQL=0
      IF(SNC(NSNSTR,5).GT.(-0.001))NQL=NQL+1
      NZER=0
      DO 1125 NP=4,NUPE
      IF(SN(NSNSTR,NP).LT.0.0001)NZER=NZER+1
1125 CONTINUE
      IF(NZER.EQ.NOCOMP)GOTO1424
      GOTO424
1424 NHE=2*NOCOMP+4
      SN(NSNSTR,NHE)=HEAT
      GOTO300
424  SN(NSNSTR,NSBCRT)=SNC(NSNSTR,6)
      KTIMES=0
427  HUN=0.
      IF(NQL.EQ.0)CALL QUAL(NSNSTR)
      DO 125 NP=4,NUPE
      IF(ABS(SN(NSNSTR,NP)).LT.0.0001)GOTO125
      CALL ENTHPY(ANS,NP-3)
      NSCRPT=NP+NOCOMP
      HUN=HUN+ANS*SN(NSNSTR,NSCRPT)
125  CONTINUE
      IF(ABS((HUN-HEAT)/HEAT).LT.0.01)GOTO300
      IF(KTIMES)425,426,426
426  KTIMES=KTIMES-1
      HUNO=HUN
      TEMP=SN(NSNSTR,NSBCRT)
      SN(NSNSTR,NSBCRT)=SNC(NSNSTR,7)

```

IF (K=1) THEN GOTO 100

IF (K=2) THEN GOTO 100

IF (K=3) THEN GOTO 100

IF (K=4) THEN GOTO 100

IF (K=5) THEN GOTO 100

IF (K=6) THEN GOTO 100

IF (K=7) THEN GOTO 100

IF (K=8) THEN GOTO 100

IF (K=9) THEN GOTO 100

IF (K=10) THEN GOTO 100

IF (K=11) THEN GOTO 100

IF (K=12) THEN GOTO 100

IF (K=13) THEN GOTO 100

IF (K=14) THEN GOTO 100

IF (K=15) THEN GOTO 100

IF (K=16) THEN GOTO 100

IF (K=17) THEN GOTO 100

IF (K=18) THEN GOTO 100

IF (K=19) THEN GOTO 100

IF (K=20) THEN GOTO 100

IF (K=21) THEN GOTO 100

IF (K=22) THEN GOTO 100

IF (K=23) THEN GOTO 100

IF (K=24) THEN GOTO 100

IF (K=25) THEN GOTO 100

IF (K=26) THEN GOTO 100

IF (K=27) THEN GOTO 100

IF (K=28) THEN GOTO 100

IF (K=29) THEN GOTO 100

IF (K=30) THEN GOTO 100

IF (K=31) THEN GOTO 100

IF (K=32) THEN GOTO 100

IF (K=33) THEN GOTO 100

IF (K=34) THEN GOTO 100

IF (K=35) THEN GOTO 100

IF (K=36) THEN GOTO 100

IF (K=37) THEN GOTO 100

IF (K=38) THEN GOTO 100

IF (K=39) THEN GOTO 100

IF (K=40) THEN GOTO 100

IF (K=41) THEN GOTO 100

IF (K=42) THEN GOTO 100

IF (K=43) THEN GOTO 100

IF (K=44) THEN GOTO 100

IF (K=45) THEN GOTO 100

IF (K=46) THEN GOTO 100

IF (K=47) THEN GOTO 100

IF (K=48) THEN GOTO 100

IF (K=49) THEN GOTO 100



```
IF (ABS(SNC(NSNSTR,7)-TEMP).LT.10.)SN(NSNSTR,NSBCRT)=TEMP+20.
```

```
GOTO427
```

```
425 SLOPE=(HUN-HUNO)/(SN(NSNSTR,NSBCRT)-TEMP)
```

```
TEMP=SN(NSNSTR,NSBCRT)
```

```
SN(NSNSTR,NSBCRT)=((HEAT-HUN)/SLOPE)+TEMP
```

```
HUNO=HUN
```

```
GOTO427
```

```
300 CONTINUE
```

```
C
```

```
C
```

```
IF PERFORMING A UNIT CALCULATION THEN RETURN TO SETUP
```

```
C
```

```
OR EQUIP
```

```
C
```

```
IF(NXB1.EQ.17)GOTO1155
```

```
DO 55 I=1,NLEGT
```

```
NSN=NSTR(I)
```

```
IF(KSFLAG(NSN).GT.0)GOTO55
```

```
NX7=0
```

```
NUP=2*NCCOMP+3
```

```
DO 59 J=3,NUP
```

```
IF(SN(NSN,J).LT.0.)NX7=NX7+1
```

```
59 CONTINUE
```

```
IF(NEB.GT.1)GOTO57
```

```
58 CONTINUE
```

```
IF(NX7.GT.0)GOTO55
```

```
C
```

```
C
```

```
FLAG STREAMS THAT HAVE BEEN TOTALLY CALCULATED TO A
```

```
C
```

```
POSITIVE NUMBER
```

```
C
```

```
IF(SN(NSN,1).LT.0.5)GOTO55
```

```
IF(SN(NSN,2).GT.1.1)GOTO55
```

```
IF(SN(NSN,2).LT.0.99)GOTO56
```

```
KSFLAG(NSN)=7
```

```
GOTO55
```

```
56 KSFLAG(NSN)=1
```

```
GOTO55
```

```
57 CONTINUE
```

```
IF(SN(NSN,NUP+1).LT.0.)NX7=NX7+1
```

```
IF(SNC(NSN,3).LT.0.)NX7=NX7+1
```

```
GOTO58
```

```
55 CONTINUE
```

```
IF(NXB1.LT.0)RETURN
```

```
1155 CONTINUE
```

```
WRITE(6,60)
```

```
60 FORMAT(1X,39HSTREAM VARIABLES OF STREAMS INVOLVED IN
```

```
117H MATERIAL BALANCE)
```

```
DO 53 I=1,NLEGT
```

```
NSN=NSTR(I)
```

```
53 WRITE(6,52)(SN(NSN,J),J=1,NSLMAX)
```

```
52 FORMAT(1X,5F14.4)
```

```
NB=NB+10
```

```
RETURN
```

```
END
```





```

SUBROUTINE SETEQ
COMMON  KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON  KEFLAG, NE, NEMAX
COMMON  KSFLAG, NS, NSMAX
COMMON  EN, NEN, NEL, NELMAX,          ENC, NENC, NECL, NE
1CLMX
COMMON  SN, NSN, NSL, NSLMAX,          SNC, NSNC, NSCL, NS
1CLMX
COMMON  KUS, NUS, NUSMAX,              KPS, NPS, NPSMAX
COMMON  NUS1, NUS2, NUS3,              KCV1, KCV2, KCV3
COMMON  KEUS, NEUS, NENUM
COMMON  KECV1, KECV2, KECV3, KECV4
COMMON  NECV1, NECV2, NECV3, NECV4
COMMON  KE1, NE1, NE1MAX,              KE2, NE2, NE2MAX
COMMON  KE3, NE3, NE3MAX,              KE4, NE4, NE4MAX
COMMON  KES, KES2, KES3
COMMON  KRET, KRET2, KRET3
COMMON  STRM, STRMC
COMMON  STRMI, STRMCI, NIN,            STRMO, STRMCO, NOUT
COMMON  DELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON  KRUN, KSETS, KCLEAN, LOOPS, NOGO
COMMON  TITLE, NOKPM, NOEN, NOENC, NCSN, NOSNC, NOKPS,
1 NSTRM
COMMON  NLX, NLY, NLXS, NLXSC
COMMON  ICONV, NEX, NECALL, NEQUIP
COMMON  ND2, ND3, ND4, ND5, ND6, ND7
COMMON  EXX, D15, D16
COMMON  I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I1
15
COMMON  J2, J3, J4, J5
COMMON  J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17, J
118, J19, J20,
1      J21, J22, J23, J24, J25, J26, J27, J28, J29, J30, J31, J3
12, J33, J34,
2      J35, J36, J37, J38, J39, J40, J41, J42, J43, J44, J45, J4
16
COMMON  J47, J48, J49, J50, J51, J52, J53, J54, J55, J56, J57, J5
18, J59, J60
COMMON  K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, K1
15, K16, K17,
1      K18, K19, K20, K21, K22, K23, K24, K25, K26, K27, K28, K2
19, K30
COMMON  L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13
COMMON  NDELS, NDELSC
COMMON  NEXEQN, NAME, KSEM
COMMON  NIPT(30), NOPT(30), LIP, LOP, NOCCMP, NUNIT(30), LUNI
1T, NB, NONOAL,
1NALLCW(15), NLEGTH, NSTR(30), COEFF(16, 16), C(16), NKSTR(16
1), NKCC(16),
2NROW, NXB1, CMPROP(15, 10), NEB, TREF
DIMENSION NEXEQN(30), NAME(30), KSEM(60, 3)
DIMENSION KPM(30, 10), KEFLAG(30), KSFLAG(60), KPMR(30
1, 12)
DIMENSION SN(60, 20), SNC(60, 10), EN(30, 10), ENC(30, 10
1)

```





```

DIMENSION KUS(60), KPS(10), KEUS(8)
DIMENSION KE1(30), KE2(30), KE3(30), KE4(30)
DIMENSION STRM(60,20), STRMC(60,10)
DIMENSION STRMI(8,20), STRMCI(8,10), STRMD(8,20), STR
1MCO(8,10)
DIMENSION DELS(20), DELSC(10)
DIMENSION TITLE(12), ICONV(15), NECALL(30)

C
C   SUBROUTINE SETEQ
C
C   THIS SUBROUTINE ATTEMPTS TO SETUP A SET OF EQUATIONS
C   TO EFFECT THE MATERIAL AND OR ENERGY BALANCES AROUND
C   THE BALANCE ENVELOPE
C
C   COEFF IS THE MATRIX OF COEFFICIENTS AND NKSTR AND
C   NKCOL ARE VECTORS CONTAINING THE ROW AND COLUMN
C   SUBSCRIPTS RESPECTIVELY OF THE UNKNOWN ELEMENTS IN THE
C   SN MATRIX
C   C IS THE VECTOR OF CONSTANTS ASSOCIATED WITH THE SET
C   OF EQUATIONS
C
C   INITIALIZE COUNTERS, INDICATORS AND COEFFICIENT MATRIX
C
      DO 100 I=1,16
      DO 100 J=1,16
100  COEFF(I,J)=0.
      LS=1
      NGAUSS=0
      NROW=0
      NCOMPO=0
      SUM=0.
      DO 1002 I=1,NLENGTH
      NSN=NSTR(I)
      NCOMP=SNC(NSN,4)+0.2
      IF(NCOMPO.LT.NCOMP)NCOMPO=NCOMP
1002  CONTINUE
      KSTR=0

C
C   ATTEMPT AN OVERALL MASS BALANCE
C
      DO 10 I=1,NLENGTH
      NSN=NSTR(I)
      NSCRPT=2*NCOMP+4
      IF(ABS(SN(NSN,NSCRPT)-10000.).LT.0.001)GOTO10
      NX3=0
      DO 22 N3=1,LIP
      IF(NSN.EQ.NIPT(N3))NX3=NX3+2
22  CONTINUE
      IF(SN(NSN,3).LT.0.)GOTO11
      IF(NX3.EQ.0)GOTO24
      SUM=SUM-SN(NSN,3)
      GOTO10
24  SUM=SUM+SN(NSN,3)
      GOTO10
11  KSTR=KSTR+1

```



```

      NKSTR(KSTR)=NSN
      NKCOL(KSTR)=3
      IF(NX3.EQ.0)GOTO21
      COEFF(1,KSTR)=1.0
      GOTO10
21  COEFF(1,KSTR)=-1.0
10  CONTINUE
      IF(KSTR.EQ.0)GOTO18
      C(1)=SUM
      NROW=NROW+1
C
C  ATTEMPT COMPONENT BALANCES
C
18  NBIG=4
      NEND=NOCOMP+3
      NX99=0
      DO 12 K3=NBIG,NEND
      IF(NGAUSS.GT.0)GOTO12
      KPRIM=KSTR
      IF(NONAL.EQ.0)GOTO1003
      NF=0
      DO 13 J=1,NONAL
      IF(K3.EQ.NALLOW(J)+3)NF=NF+1
13  CONTINUE
      IF(NF.GT.0)GOTO12
1003 NQ=0
      NF=0
      NX9=0
      SUM=C.
      NX=0
      NROW=NROW+1
      DO 16 I=1,NLENGH
      IF(NX9.GT.0)GOTO16
      NSN=NSTR(I)
      NX3=0
      DO 28 N3=1,LIP
      IF(NSN.EQ.NIPT(N3))NX3=NX3+2
28  CONTINUE
      IF(SN(NSN,K3).LT.0.)GOTO17
      GOTO26
17  CONTINUE
      IF(SN(NSN,3).LT.0.)GOTO216
      KSTR=KSTR+1
      NKSTR(KSTR)=NSN
      NKCOL(KSTR)=K3
      IF(NX3.EQ.0)GOTO29
      COEFF(NROW,KSTR)=SN(NSN,3)
      GOTO16
216  NST1=K3+NOCOMP
      IF(SN(NSN,NST1).GT.0.)GOTO1216
      GOTO1217
1216 IF(NX3.GT.0)GOTO1218
      SUM=SUM+SN(NSN,NST1)
      GOTO16
1218 SUM=SUM-SN(NSN,NST1)

```





```

      GOTO16
1217 NX9=NX9+1
C
C      TRY NEXT COMPONENT BALANCE BUT IF HAVE TO DO ENERGY
C      BALANCE MAY HAVE TO DISCARD OVERALL MASS BALANCE
C      EQUATION AND PERFORM ONLY COMPONENT BALANCES
C
      NX99=NX99-1
      GOTO16
29 COEFF(NROW,KSTR)=-SN(NSN,3)
      GOTO16
26 CONTINUE
      IF(SN(NSN,3).LT.0.)GOTO30
      IF(NX3.EQ.0)GOTO31
      SUM=SUM-SN(NSN,3)*SN(NSN,K3)
      GOTO16
31 SUM=SUM+SN(NSN,3)*SN(NSN,K3)
      GOTO16
30 CONTINUE
      KNOTE=0
      DO 33 K=1,KSTR
      IF(NSN.EQ.NKSTR(K))GOTO34
      GOTO33
34 CONTINUE
      IF(3.EQ.NKCOL(K))KNOTE=K
33 CONTINUE
      IF(KNOTE.EQ.0)GOTO132
133 IF(NX3.EQ.0)GOTO32
      COEFF(NROW,KNOTE)=SN(NSN,K3)
      GOTO16
32 COEFF(NROW,KNOTE)=-SN(NSN,K3)
      GOTO16
132 KSTR=KSTR+1
      NKSTR(KSTR)=NSN
      NKCOL(KSTR)=3
      KNOTE=KSTR
      GOTO133
16 CONTINUE
      IF(NX9.GT.0)GOTO217
      C(NROW)=SUM
      NX11=0
      KEHC=0
      NX=0
      NUP=NROW-1
      IF(NUP.EQ.0)GOTO219
C
C      CHECK FOR LINEAR DEPENDENCE OF OF ANY TWO EQUATIONS
C
      DO 711 ICHECK=1,NUP
      IF(NX.GT.0)GOTO711
      NX1=0
      IF(ABS(C(NROW)).LT.0.0000001)GOTO713
      ACHEK=C(ICHECK)/C(NROW)
      GOTO714
713 IF(ABS(C(ICHECK)).LT.0.0000001)KEHC=1

```





```

    ACHEK=10000000.
714 DO 712 JCHECK=1,KSTR
    IF(ABS(COEFF(NROW,JCHECK)).LT.0.0000001)GOTO730
    ACHEK1=COEFF(ICHECK,JCHECK)/COEFF(NROW,JCHECK)
    IF(KEHC.GT.0)ACHEK=ACHEK1
    KEHC=0
732 IF(ABS(ACHEK).LT.0.0000001)GOTO715
    IF(ABS((ACHEK-ACHEK1)/ACHEK).LT.0.00001)NX1=NX1+1
    GOTO716
715 IF(ABS(ACHEK-ACHEK1).LT.0.0000001)NX1=NX1+1
716 IF(ICHECK.GT.1)GOTO712
    CFF=COEFF(NROW,JCHECK)
    IF(ABS(CFF).LT.0.000001)GOTO712
    NX11=NX11+1
712 CONTINUE
    IF(NX1.EQ.KSTR)NX=NX+1
    IF(NX11.EQ.0)NX=NX+1
711 CONTINUE
    IF(NX.GT.0)GOTO217
    GOTO219
730 IF(ABS(COEFF(ICHECK,JCHECK)).LT.0.000001)GOTO731
    ACHEK1=10000000.
    GOTO732
731 IF(KEHC.EQ.1)GOTO733
    ACHEK1=ACHEK
    GOTO732
733 ACHEK=1.
    ACHEK1=1.
    GOTO732
217 NROW=NROW-1
C
C   ELIMINATE LAST COMPONENT BALANCE EQUATION
C
    KSTR=KPRIM
    DO 1010 I=1,16
1010 COEFF(NROW+1,I)=0.
219 CONTINUE
C
C   CHECK TO SEE IF ENOUGH EQUATIONS OBTAINED
C
    IF(KSTR.GT.NCOMPO+1)GOTO61
    IF(NROW.EQ.NCOMPO)NGAUSS=NGAUSS+1
    IF(NROW.EQ.KSTR)NGAUSS=NGAUSS+1
12 CONTINUE
    IF(NROW.EQ.KSTR)GOTO35
    IF(NEB.EQ.1)GOTO61
    IF(NX99.LT.0)GOTO80
    IF(KSTR-NROW.EQ.1)GOTO36
61 WRITE(6,37)
C
C   UNABLE TO FIND SUITABLE SET OF EQUATIONS RETURN
C   CONTROL TO SEQCAL
C
37 FORMAT(1X,15HUNABLE TO SOLVE)
    NXB1=NXB1-1

```





```

      RETURN
80  NX9=0
C
C  ATTEMPT COMPONENT BALANCES ONLY AS A PRELUDE TO
C  MATERIAL AND ENERGY BALANCES IN CONJUNCTION
C
      K3=2*NOCOMP+4
      DO 81 I=1,NLEGT
      NSN=NSTR(I)
      IF(ABS(SN(NSN,K3)+1.) .LT. 0.0001) NX9=NX9+1
81  CONTINUE
C
C  IF ALL TEMPERATURES AND HEAT FLOWS INVOLVED IN THE
C  BALANCE CALCULATION ARE NOT GIVEN THEN RETURN CONTROL
C  TO SEQCAL
C
      IF(NX9.EQ.0) GOTO83
      GOTO61
83  KSTR=0
      DO 1011 I=1,16
      DO 1011 J=1,16
1011 COEFF(I,J)=0.
      NROW=0
      NGAUSS=0
      NUL=NOCOMP+3
      DO 84 K3=4,NUL
      IF(NGAUSS.GT.0) GOTO84
      KPRIM=KSTR
      SUM=0.
      IF(NONOAL.EQ.0) GOTO1004
      NF=0
      DO 85 J=1,NONOAL
      IF(K3.EQ.NALLOW(J)+3) NF=NF+1
85  CONTINUE
      IF(NF.GT.0) GOTO84
1004 NROW=NROW+1
      DO 86 I=1,NLEGT
      NSN=NSTR(I)
      NX3=0
      DO 87 N5=1,LIP
      IF(NSN.EQ.NIPT(N5)) NX3=NX3+2
87  CONTINUE
      IF(SN(NSN,3).LT.0.) GOTO88
      IF(SN(NSN,K3).LT.0.) GOTO90
      IF(NX3.GT.0) GOTO89
      SUM=SUM+SN(NSN,K3)*SN(NSN,3)
      GOTO86
89  SUM=SUM-SN(NSN,K3)*SN(NSN,3)
86  CONTINUE
      C(NROW)=SUM
      NUP=NROW-1
      IF(NUP.EQ.0) GOTO784
C
C  CHECK FOR LINEAR DEPENDENCE OF OF ANY TWO EQUATIONS
C

```



```

      NX11=0
      KEHC=0
      NX=0
      DO 717 ICHECK=1,NUP
      IF(NX.GT.0)GOTO717
      NX1=0
      IF(ABS(C(NROW)).LT.0.0000001)GOTO718
      ACHEK=C(ICHECK)/C(NROW)
      GOTO719
718  IF(ABS(C(ICHECK)).LT.0.0000001)KEHC=1
      ACHEK=10000000.
719  DO 720 JCHECK=1,KSTR
      IF(ABS(COEFF(NROW,JCHECK)).LT.0.0000001)GOTO734
      ACHEK1=COEFF(ICHECK,JCHECK)/COEFF(NROW,JCHECK)
      IF(KEHC.GT.0)ACHEK=ACHEK1
      KEHC=0
736  IF(ABS(ACHEK).LT.0.0000001)GOTO721
      IF(ABS((ACHEK-ACHEK1)/ACHEK).LT.0.00001)NX1=NX1+1
      GOTO722
721  IF(ABS(ACHEK-ACHEK1).LT.0.0000001)NX1=NX1+1
722  IF(ICHECK.GT.1)GOTO720
      CFF=COEFF(NROW,JCHECK)
      IF(ABS(CFF).LT.0.0000001)GOTO720
      NX11=NX11+1
720  CONTINUE
      IF(NX1.EQ.KSTR)NX=NX+1
      IF(NX11.EQ.0)NX=NX+1
717  CONTINUE
      IF(NX.EQ.0)GOTO784
      NROW=NROW-1
      KSTR=KPRIM
      DO 1012 I=1,16
1012  COEFF(NROW+1,I)=0.
      GOTO784
734  IF(ABS(COEFF(ICHECK,JCHECK)).LT.0.0000001)GOTO735
      ACHEK1=10000000.
      GOTO736
735  IF(KEHC.EQ.1)GOTO737
      ACHEK1=ACHEK
      GOTO736
737  ACHEK=1.
      ACHEK1=1.
      GOTO736
784  IF(NROW.EQ.NCOMPO)NGAUSS=NGAUSS+1
      IF(KSTR.EQ.NROW)NGAUSS=NGAUSS+1
      IF(KSTR.GT.NCOMPO+1)GOTO61
84  CONTINUE
      GOTO94
88  CONTINUE
      IF(SN(NSN,K3).LT.0.)GOTO91
      KNOTE=0
      IF(KSTR.EQ.0)GOTO1007
      DO 1005 KI5=1,KSTR
      IF(NSN.EQ.NKSTR(KI5))GOTO1006
      GOTO1005

```





```

1006 IF(3.EQ.NKCOL(KI5))KNOTE=KI5
1005 CONTINUE
      IF(KNOTE.EQ.0)GOTO1007
      IF(NX3.GT.0)GOTO1008
      COEFF(NROW,KNOTE)=-SN(NSN,K3)
      GOTO86
1008 COEFF(NROW,KNOTE)=SN(NSN,K3)
      GOTO86
1007 KSTR=KSTR+1
      NKSTR(KSTR)=NSN
      NKCOL(KSTR)=3
      IF(NX3.GT.0)GOTO92
      COEFF(NROW,KSTR)=-SN(NSN,K3)
      GOTO86
92 COEFF(NROW,KSTR)=SN(NSN,K3)
      GOTO86
91 CONTINUE
      NCOLM=K3+NOCOMP
      IF(SN(NSN,NCOLM).LT.0.)GOTO191
      IF(NX3.GT.0)GOTO193
      SUM=SUM+SN(NSN,NCOLM)
      GOTO86
193 SUM=SUM-SN(NSN,NCOLM)
      GOTO86
191 KSTR=KSTR+1
      NKSTR(KSTR)=NSN
      NKCOL(KSTR)=K3+NOCOMP
      IF(NX3.GT.0)GOTO93
      COEFF(NROW,KSTR)=-1.0
      GOTO86
93 COEFF(NROW,KSTR)=+1.0
      GOTO86
90 KSTR=KSTR+1
      NKSTR(KSTR)=NSN
      NKCOL(KSTR)=K3
      IF(NX3.GT.0)GOTO95
      COEFF(NROW,KSTR)=-SN(NSN,3)
      GOTO86
95 COEFF(NROW,KSTR)=SN(NSN,3)
      GOTO86
94 CONTINUE
      IF((KSTR-NROW).EQ.0)GOTO35
      IF((KSTR-NROW).EQ.1)GOTO36
      GOTO61

```

```

C
C   AN ENERGY BALANCE IS ATTEMPTED IF AND ONLY IF ONE MORE
C   EQUATION IS REQUIRED TO SOLVE THE BALANCE
C

```

```

36 LS=LS-1
      IF(NEB.EQ.1)GOTO61
      K3=2*NOCOMP+4
      DO 38 I=1,NLEGT
      NSN=NSTR(I)
      NZER=0
      NUPPER=NOCOMP+3

```





```

C
C      NO TEMPERATURES, PRESSURES, QUALITIES OR HEAT FLOWS
C      CAN BE UNKNOWN OR CONTROL IS RETURNED TO SEQCAL
C
      DO 1138 IHT=4,NUPPER
      IF(ABS(SN(NSN,IHT)).LT.0.0001)NZER=NZER+1
1138  CONTINUE
      IF(NZER.EQ.NOCOMP)GOTO1136
      IF(SN(NSN,K3).LT.0.)GOTO61
      IF(SNC(NSN,3).LT.0.)GOTO61
      IF(SNC(NSN,5).LT.0.)GOTO61
      GOTO38
1136  IF(ABS(SN(NSN,K3)+1.).LT.0.0001)GOTO61
      38  CONTINUE
      NROW=NRUW+1
      SUMH=0.
      NUPE=NOCOMP+3
      DO 42 J=1,NLEGT
      NSN=NSTR(J)
      NX3=0
      DO 124 N3=1,LIP
      IF(NSN.EQ.NIPT(N3))NX3=NX3+2
124  CONTINUE
      SUM2=0.
      NX4=0
      NZER=0
      IF(ABS(SN(NSN,K3)-10000.).LT.0.001)GOTO42
      DO 109 NP=4,NUPE
      IF(SN(NSN,NP).LT.0.001)GOTO1109
      CALL ENTHPY(SUM3,NP-3)
      GOTO111
1109  NZER=NZER+1
      111  CONTINUE
      IF(NZER.EQ.NOCOMP)GOTO518
      IF(SN(NSN,3).LT.0.)GOTO110
      IF(SN(NSN,NP).LT.0.)GOTO112
      IF(NX3.EQ.0)GOTO113
      SUMH=SUMH-SN(NSN,NP)*SN(NSN,3)*SUM3
      GOTO109
518  NHE=2*NOCOMP+4
      IF(NX3.GT.0)GOTO1518
      SUMH=SUMH+SN(NSN,NHE)
      GOTO109
1518  SUMH=SUMH-SN(NSN,NHE)
      GOTO109
      113  SUMH=SUMH+SN(NSN,NP)*SN(NSN,3)*SUM3
      GOTO109
      110  CONTINUE
      IF(SN(NSN,NP).LT.0.)GOTO102
      DO 114 NI=1,KSTR
      IF(NSN.EQ.NKSTR(NI))GOTO115
      GOTO114
115  CONTINUE
      IF(3.EQ.NKCOL(NI))KNOTE=NI
114  CONTINUE

```





```

      IF(NX3.EQ.0)GOTO116
      SUM2=SUM2+SN(NSN,NP)*SUM3
      COEFF(NROW,KNOTE)=SUM2
      GOTO109
116  SUM2=SUM2-SN(NSN,NP)*SUM3
      COEFF(NROW,KNOTE)=SUM2
      GOTO109
102  CONTINUE
      NCOLM=NP+NOCOMP
      IF(SN(NSN,NCOLM).LT.0.)GOTO1102
      IF(NX3.EQ.0)GOTO1119
      SUMH=SUMH-SN(NSN,NCOLM)*SUM3
      GOTO109
1119 SUMH=SUMH+SN(NSN,NCOLM)*SUM3
      GOTO109
1102 DO 117 K=1,KSTR
      IF(NSN.EQ.NKSTR(K))GOTO118
      GOTO117
118  CONTINUE
      IF(NP+NOCOMP.EQ.NKSTR(K))KNOTE=K
117  CONTINUE
      IF(NX3.EQ.0)GOTO119
      COEFF(NROW,KNOTE)=SUM3
      GOTO109
119  COEFF(NROW,KNOTE)=-SUM3
      GOTO109
112  CONTINUE
      DO 120 K=1,KSTR
      IF(NSN.EQ.NKSTR(K))GOTO121
      GOTO120
121  CONTINUE
      IF(NP.EQ.NKCOL(K))KNOTE=K
120  CONTINUE
      IF(NX3.EQ.0)GOTO122
      COEFF(NROW,KNOTE)=SN(NSN,3)*SUM3
      GOTO109
122  COEFF(NROW,KNOTE)=-SN(NSN,3)*SUM3
109  CONTINUE
      42  CONTINUE
      C(NROW)=SUMH
      35  CONTINUE
C
C      A SUITABLE SET OF EQUATIONS HAS BEEN SETUP AND
C      SOLUTION WILL BE ATTEMPTED BY SUBROUTINE SIMCAL
C
      WRITE(6,245)
245  FORMAT(//1X,18HCOEFFICIENT MATRIX/)
      NSC=0
      DO 243 I=1,NROW
      IF(ABS(C(NROW)).LT.0.0001)NSC=NSC+1
243  WRITE(6,244)(COEFF(I,J),J=1,NROW)
244  FORMAT(8(2X,F10.4))
      IF(NSC.EQ.NROW)GOTO61
      WRITE(6,248)
248  FORMAT(//8X,8HUNKNOWN,10X,19HVECTOR OF CONSTANTS/)

```





```
DO 246 I=1,NROW  
246 WRITE(6,247)NKSTR(I),NKCOT(I),C(I)  
247 FORMAT(6X,3H5N(,I3,1H,,I3,1H),12X,F20.4)  
CALL SIMCAL  
RETURN  
END
```



```

SUBROUTINE SIMCAL
COMMON  KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON  KEFLAG, NE, NEMAX
COMMON  KSFLAG, NS, NSMAX
COMMON  EN, NEN, NEL, NELMAX,          ENC, NENC, NECL, NE
1CLMX
COMMON  SN, NSN, NSL, NSLMAX,          SNC, NSNC, NSCL, NS
1CLMX
COMMON  KUS, NUS, NUSMAX,              KPS, NPS, NPSMAX
COMMON  NUS1, NUS2, NUS3,              KCV1, KCV2, KCV3
COMMON  KEUS, NEUS, NENUM
COMMON  KECV1, KECV2, KECV3, KECV4
COMMON  NECV1, NECV2, NECV3, NECV4
COMMON  KE1, NE1, NE1MAX,              KE2, NE2, NE2MAX
COMMON  KE3, NE3, NE3MAX,              KE4, NE4, NE4MAX
COMMON  KES, KES2, KES3
COMMON  KRET, KRET2, KRET3
COMMON  STRM, STRMC
COMMON  STRM1, STRMC1, NIN,            STRMO, STRMCC, NOUT
COMMON  DELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON  KRUN, KSETS, KCLEAN, LOOPS, NOGO
COMMON  TITLE, NCKPM, NOEN, NOENC, NCSN, NOSNC, NOKPS,
1 NSTRM
COMMON  NLX, NLY, NLXS, NLXSC
COMMON  ICONV, NEX, NECALL, NEQUIP
COMMON  ND2, ND3, ND4, ND5, ND6, ND7
COMMON  EXX, D15, D16
COMMON  I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I1
15
COMMON  J2, J3, J4, J5
COMMON  J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17, J
118, J19, J20,
1 J21, J22, J23, J24, J25, J26, J27, J28, J29, J30, J31, J3
12, J33, J34,
2 J35, J36, J37, J38, J39, J40, J41, J42, J43, J44, J45, J4
16
COMMON  J47, J48, J49, J50, J51, J52, J53, J54, J55, J56, J57, J5
18, J59, J60
COMMON  K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, K1
15, K16, K17,
1 K18, K19, K20, K21, K22, K23, K24, K25, K26, K27, K28, K2
19, K30
COMMON  L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13
COMMON  NDELS, NDELSC
COMMON  NEXEQN, NAME, KSEM
COMMON  NIPT(30), NOPT(30), LIP, LOP, NCCOMP, NUNIT(30), LUNI
1T, NB, NONOAL,
1NALLOW(15), NLEGTN, NSTR(30), COEFF(16,16), C(16), NKSTR(16
1), NKCOL(16),
2NROW, NXB1, CMPRC(15,10), NEB, TREF
DIMENSION NEXEQN(30), NAME(30), KSEM(60,3)
DIMENSION KPM(30,10), KEFLAG(30), KSFLAG(60), KPMR(30
1,12)
DIMENSION SN(60,20), SNC(60,10), EN(30,10), ENC(30,10
1)

```





```

      DIMENSION KUS(60), KPS(10), KEUS(8)
      DIMENSION KF1(30), KE2(30), KE3(30), KE4(30)
      DIMENSION STRM(60,20), STRMC(60,10)
      DIMENSION STRMI(8,20), STRMCI(8,10), STRMD(8,20), STR
1MCC(8,10)
      DIMENSION DELS(20), DELSC(10)
      DIMENSION TITLE(12), ICONV(15), NECALL(30), NRO(16)

```

C  
C  
C  
C  
C

```

      SUBROUTINE SIMCAL

```

```

      NROW EQUATIONS ARE SOLVED BY GAUSSIAN ELIMINATION

```

```

114 NTIME=0
      DO 115 I=1,NROW
115 NRO(I)=I
      14 NRINT=0
      DO 10 I=1,NROW
      BA=CCEFF(I,I)
      IF(ABS(BA).LT.0.005)NRINT=I
10 CONTINUE
      IF(NRINT.EQ.0)GOTO11
      NTIME=NTIME+1
      IF(NTIME.EQ.NROW)GOTO30
      NR=0
      DO 12 I=1,NROW
      CA=CCEFF(I,NRINT)
      IF(ABS(CA).LT.0.005)GOTO12
      IF(NRO(I).EQ.NRINT)GOTO117
      GOTO116
117 IF(NRO(NRINT).EQ.I)GOTO12
116 NR=I
      12 CONTINUE
      IF(NR.EQ.0)GOTO30
      DO 13 I=1,NROW
      STR=CCEFF(NRINT,I)
      COEFF(NRINT,I)=COEFF(NR,I)
      COEFF(NR,I)=STR
13 CONTINUE
      D=C(NRINT)
      C(NRINT)=C(NR)
      C(NR)=D
      NTEM=NRO(NR)
      NRO(NR)=NRO(NRINT)
      NRO(NRINT)=NTEM
      GOTO14
11 CONTINUE
      NUPPER=NROW-1
      NZEROD=0
      DO 15 NR=1,NUPPER
      IF(NZEROD.GT.0)GOTO15
      NJ=NR+1
      DO 16 I=NJ,NROW
      IF(NZEROD.GT.0)GOTO16
      C(I)=C(I)-COEFF(I,NR)*C(NR)/COEFF(NR,NR)
      DO 17 J=NJ,NROW

```





```

17 COEFF(I,J)=COEFF(I,J)-COEFF(I,NR)*COEFF(NR,J)/COEFF(NR
  1,NR)
  IF(ABS(COEFF(I,I)).LT.0.005)NZEROD=I
16 CONTINUE
  DO 18 JG=NJ,NROW
18 COEFF(NR,JG)=COEFF(NR,JG)/COEFF(NR,NR)
  C(NR)=C(NR)/COEFF(NR,NR)
  COEFF(NR,NR)=1.
  NUPPER=NROW
  IF(NZEROD.GT.0)NUPPER=NZEROD
  DO 37 I=NJ,NUPPER
  DO 37 J=1,NR
37 COEFF(I,J)=0.
15 CONTINUE
  IF(NZEROD.GT.0)GOTO114
  DO 19 I=1,NROW
    SUM=C.
    NK=NROW+1-I
    N=NKSTR(NK)
    M=NKCOL(NK)
    NJ=NK+1
    IF(NJ.GT.NROW)GOTO22
    DO 20 J=NJ,NROW
      N1=NKSTR(J)
      M1=NKCOL(J)
20 SUM=SUM+COEFF(NK,J)*SN(N1,M1)
      SN(N,M)=C(NK)-SUM
      GOTO21
22 SN(N,M)=C(NK)/COEFF(NK,NK)
21 CONTINUE
    WRITE(6,23)N,M,SN(N,M)
23 FORMAT(1X,3H$N(,I5,1H,,I5,1H),5X,1H=,5X,F10.2)
    IF(SN(N,M).LT.C.)SN(N,M)=0.
19 CONTINUE
  RETURN
30 NCALC=0
  DO 32 I=1,NROW
    NELMT=0
    NZS=0
    DO 31 J=1,NROW
      IF(ABS(COEFF(I,J)).LT.0.0000001)NZS=NZS+1
      IF(ABS(COEFF(I,J)).GT.0.0099)NELMT=J
31 CONTINUE
      IF(NZS.EQ.(NROW-1))GOTO33
      GOTO32
33 NN=NKSTR(NELMT)
      MM=NKCOL(NELMT)
      SN(NN,MM)=C(I)/COEFF(I,NELMT)
      IF(SN(NN,MM).LT.0.)SN(NN,MM)=0.
      NCALC=-100
32 CONTINUE
      IF(NCALC.LT.0)RETURN
      WRITE(6,34)
34 FORMAT(44H UNABLE TO SOLVE THESE EQUATIONS BY GAUSSIAN
    112H ELIMINATION)

```



C  
C ANOTHER METHOD OF SOLUTION SHOULD BE PROVIDED HERE  
C IN CASE OF THE FAILURE OF THIS SUBROUTINE TO SOLVE THE  
C SET OF EQUATIONS

C  
NXBI=NXBI-1

RETURN

END







```

SUBROUTINE ENTHPY(ANS,NCP)
COMMON KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON KEFLAG, NE, NEMAX
COMMON KSFLAG, NS, NSMAX
COMMON EN, NEN, NEL, NELMAX, ENC, NENC, NECL, NE
1CLMX
COMMON SN, NSN, NSL, NSLMAX, SNC, NSNC, NSCL, NS
1CLMX
COMMON KUS, NUS, NUSMAX, KPS, NPS, NPSMAX
COMMON NUS1, NUS2, NUS3, KCV1, KCV2, KCV3
COMMON KEUS, NEUS, NENUM
COMMON KECV1, KECV2, KECV3, KECV4
COMMON NECV1, NECV2, NECV3, NECV4
COMMON KE1, NE1, NE1MAX, KE2, NE2, NE2MAX
COMMON KE3, NE3, NE3MAX, KE4, NE4, NE4MAX
COMMON KES, KES2, KES3
COMMON KRET, KRET2, KRET3
COMMON STRM, STRMC
COMMON STRM1, STRMC1, NIN, STRMO, STRMCO, NOUT
COMMON DELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON KRUN, KSETS, KCLEAN, LCOPS, NOGO
COMMON TITLE, NOKPM, NOEN, NOENC, NCSN, NOSNC, NOKPS,
1 NSTRM
COMMON NLX, NLY, NLXS, NLXSC
COMMON ICONV, NEX, NECALL, NEQUIP
COMMON ND2, ND3, ND4, ND5, ND6, ND7
COMMON EXX, D15, D16
COMMON I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I1
15
COMMON J2,J3,J4,J5
COMMON J6, J7,J8,J9,J10,J11,J12,J13,J14,J15,J16,J17,J
118,J19,J20,
1 J21,J22,J23,J24,J25,J26,J27,J28,J29,J30,J31,J3
12,J33,J34,
2 J35,J36,J37,J38,J39,J40,J41,J42,J43,J44,J45,J4
16
COMMON J47,J48,J49,J50,J51,J52,J53,J54,J55,J56,J57,J5
18,J59,J60
COMMON K2,K3,K4,K5,K6,K7,K8,K9,K10,K11,K12,K13,K14,K1
15,K16,K17,
1 K18,K19,K20,K21,K22,K23,K24,K25,K26,K27,K28,K2
19,K30
COMMON L2,L3,L4,L5,L6,L7,L8,L9,L10,L11,L12,L13
COMMON NDELS,NDELSC
COMMON NEXEQN, NAME,KSEM
COMMON NIPT(30),NOPT(30),LIP,LOP,NCCOMP,NUNIT(30),LUNI
1T,NB,NONOAL,
1NALLOW(15),NLEGTN,NSTR(30),COEFF(16,16),C(16),NKSTR(16
1),NKCOT(16),
2NROW,NXB1,CMPROP(15,10),NEB,TREF
COMMON NEXEQN(30), NAME(30),KSEM(60,3)
COMMON KPM(30,10), KEFLAG(30), KSFLAG(60), KPMR(30
1,12)
COMMON SN(60,20) ,SNC(60,10), EN(30,10), ENC(30,10
1)

```





```

DIMENSION KUS(60), KPS(10), KEUS(8)
DIMENSION KEL(30), KE2(30), KE3(30), KE4(30)
DIMENSION STRM(60,20), STRMC(60,10)
DIMENSION STRMI(8,20), STRMCI(8,10), STRMO(8,20), STR
IMCO(8,10)
DIMENSION DELS(20), DELSC(10)
DIMENSION TITLE(12), ICONV(15), NECALL(30)
REAL K
REAL LHV

```

C

C SUBROUTINE ENTHPY

C

C REFERENCE STATE FOR COMPONENTS IS SATURATED LIQUID AT  
 C TREF DEGREES RANKINE UNLESS AT THIS TEMPERATURE THE  
 C PURE COMPONENT EXISTS AS A GAS AT ONE ATMOSPHERE  
 C PRESSURE

C

C MIXING EFFECTS NEGLECTED.

C

C ELEMENTS OF CMPROP MATRIX CONTAIN-

C

C ELEMENT PHYSICAL PROPERTY

C

C 1 LATENT HEAT OF VAPOURIZATION 'NBP

C

C 2 NORMAL BOILING POINT DEGREES RANKINE

C

C 3 HEAT CAPACITY FOR GAS OR FIRST CONSTANT

C

C IN HEAT CAPACITY EQUATION

C

C 4 SECOND TERM OF HEAT CAPACITY EQUATION

C

C TIMES 10.\*\*2

C

C 5 THIRD CONSTANT OF HEAT CAPACITY EQUATION

C

C TIMES 10.\*\*5

C

C 6 HEAT CAPACITY FOR LIQUID

C

C 7 VAPOUR PRESSURE(PSIA)'TREF

C

C 8 MOLECULAR WEIGHT

C

C 9 CRITICAL TEMPERATURE DEGREES RANKINE

C

C 10 CRITICAL PRESSURE PSIA

C

C HEAT CAPACITY EQUATION OF FORM

C

C  $CP=A+B*T+C*T**2$ 

C

C IF VAPOUR PRESSURE AT TREF IS GREATER THAN ATMOSPHERIC  
 C PRESSURE THEN THE REFERENCE STATE FOR THAT COMPONENT  
 C IS VAPOUR AT 14.7 PSIA AND TREF DEGREES RANKINE,  
 C OTHERWISE IT IS SATURATED LIQUID AT TREF

C

C THE EQUATION OF STATE CHOSEN TO EVALUATE ENTHALPY  
 C ENTHALPY DEPARTURES VAN DER WAALS EQUATION OF STATE

C

C WATSON,S EMPIRICAL CORRELATION IS USED TO EVALUATE  
 C THE LATENT HEAT OF VAPOURIZATION AT TREF FROM THAT  
 C WHICH IS GIVEN AT NBP

C

C PRESSURE IS STORED IN SNC(NSN,3)

C

C TEMPERATURE IN SN(NSN,NSLMAX)

C

C QUALITY IN SNC(NSN,5) AND VAPOUR RATIO IN SNC(NSN,8)

C

C ALL FOUR OF THESE MUST BE KNOWN BEFORE THIS SUBROUTINE



```

C      IS ENTERED
C
C      HEAT CAPACITIES ARE GIVEN AT ONE ATM PRESSURE
C
      R=10.73/CMPROP(NCP,8)
      TB=CMPROP(NCP,2)
      LHV=CMPROP(NCP,1)*((1.-TREF/CMPROP(NCP,9))/(1.-TB/CMPR
1CP(NCP,9)))*
1*.38
      CPG=CMPROP(NCP,3)
      CPG1=CMPROP(NCP,4)
      CPG2=CMPROP(NCP,5)
      CPL=CMPROP(NCP,6)
      CONV=144.*.0012856
      A=27.*R**2*CMPROP(NCP,9)**2/(64.*CMPROP(NCP,10))
      B=R*CMPROP(NCP,9)/(8.*CMPROP(NCP,10))
      KPHAS=0
100  IF(SNC(NSN,5).GT.0.01)GOTO518
      GOTO519
518  IF(SNC(NSN,5).LT.0.99)GOTO424
519  IF(CMPROP(NCP,7).GT.14.7)GOTO418
1418 IF(SNC(NSN,5).LT.0.01)GOTO416
      GOTO417
416  ANS=CPL*(SN(NSN,K3)-TREF)
      IF(KPHAS.EQ.0)RETURN
      GOTO425
418  NUM=2
      IF(NEB.EQ.3)GOTO1418
      T=SN(NSN,K3)
      P=14.7
      GOTO500
420  ANS=PRCORR
      GOTO423
417  ANS=C.
      P=CMPROP(NCP,7)
      T=TREF
      NUM=1
      GOTO500
421  ANS=ANS+PRCORR+LHV
      P=14.7
      T=TREF
      NUM=4
      GOTO 500
430  ANS=ANS-PRCORR
      P=14.7
      T=SN(NSN,K3)
      NUM=5
      GOTO 500
431  ANS=ANS+PRCORR
423  P=SNC(NSN,3)
      T=SN(NSN,K3)
      NUM=3
      GOTO500
422  ANS=ANS+CPG*(SN(NSN,K3)-TREF)-PRCORR
1+CPG1*(SN(NSN,K3)**2-TREF**2)/200.

```





```

2+CPG2*(SN(NSN,K3)**3-TREF**3)/300000.
  IF(KPHAS.EQ.0)RETURN
  GOTO425
500 VO=R*T/P
  IF(NEB.EQ.2)GOTO501
  PDCORR=0.
  GOTO(421,420,422,430,431),NUM
501 FOFVC=(P+A/VO**2)*(VO-B)-R*T
  FPRIM=P-A/VO**2+2.*A*B/VO**3
  V=VO-FOFVO/FPRIM
  IF(((V-VO)/VO)-0.01)502,502,503
503 VO=V
  GOTO501
502 PRCORR=(R*T-P*V+A/V)*CONVF
  GOTO(421,420,422,430,431),NUM
424 KPHAS=KPHAS+1
  Q=SNC(NSN,5)
  P=SNC(NSN,3)
  T=SN(NSN,K3)
  I=NCP
  CALL EQUILK(I,K,T,P)
  RATIO=K*SN(NSN,8)/(1.-SN(NSN,8))
  R=RATIO/(RATIO+1.)
  SNC(NSN,5)=1.0
  GOTO100
425 IF(KPHAS.GT.1)GOTO426
  HV=ANS
  SNC(NSN,5)=0.
  KPHAS=KPHAS+1
  GOTO100
426 HL=ANS
  ANS=R*HV+(1.-R)*HL
  SNC(NSN,5)=Q
  RETURN
  END

```





```

SUBROUTINE EQUILK(I,K,T,P)
COMMON  KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON  KEFLAG, NE, NEMAX
COMMON  KSFLAG, NS, NSMAX
COMMON  EN, NEN, NEL, NELMAX,          ENC, NENC, NECL, NE
1CLMX
COMMON  SN, NSN, NSL, NSLMAX,          SNC, NSNC, NSCL, NS
1CLMX
COMMON  KUS, NUS, NUSMAX,              KPS, NPS, NPSMAX
COMMON  NUS1, NUS2, NUS3,              KCV1, KCV2, KCV3
COMMON  KEUS, NEUS, NENUM
COMMON  KECV1, KECV2, KECV3, KECV4
COMMON  NECV1, NECV2, NECV3, NECV4
COMMON  KE1, NE1, NE1MAX,              KE2, NE2, NE2MAX
COMMON  KE3, NE3, NE3MAX,              KE4, NE4, NE4MAX
COMMON  KES, KES2, KES3
COMMON  KRET, KRET2, KRET3
COMMON  STRM, STRMC
COMMON  STRM1, STRMC1, NIN,            STRMO, STRMCO, NOUT
COMMON  DELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON  KRUN, KSETS, KCLEAN, LOOPS, NOGO
COMMON  TITLE, NOKPM, NOEN, NOENC, NCSN, NOSNC, NOKPS,
1 NSTRM
COMMON  NLX, NLY, NLXS, NLXSC
COMMON  ICONV, NEX, NCALL, NEQUIP
COMMON  ND2, ND3, ND4, ND5, ND6, ND7
COMMON  EXX, D15, D16
COMMON  I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I1
15
COMMON  J2, J3, J4, J5
COMMON  J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17, J
118, J19, J20,
1 J21, J22, J23, J24, J25, J26, J27, J28, J29, J30, J31, J3
12, J33, J34,
2 J35, J36, J37, J38, J39, J40, J41, J42, J43, J44, J45, J4
16
COMMON  J47, J48, J49, J50, J51, J52, J53, J54, J55, J56, J57, J5
18, J59, J60
COMMON  K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, K1
15, K16, K17,
1 K18, K19, K20, K21, K22, K23, K24, K25, K26, K27, K28, K2
19, K30
COMMON  L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13
COMMON  NDELS, NDELSC
COMMON  NEXEQN, NAME, KSEM
COMMON  NIPT(30), NOPT(30), LIP, LOP, NCCOMP, NUNIT(30), LUNI
IT, NB, NONOAL,
INALLOW(15), NLEGT, NSTR(30), COEFF(16,16), C(16), NKSTR(16
1), NKCOL(16),
2NROW, NXB1, CMPROP(15,10), NEB, TREF
DIMENSION NEXEQN(30), NAME(30), KSEM(60,3)
DIMENSION KPM(30,10), KEFLAG(30), KSFLAG(60), KPMR(30
1,12)
DIMENSION SN(60,20), SNC(60,10), EN(30,10), ENC(30,10
1)

```





```

DIMENSION KUS(60), KPS(10), KEUS(8)
DIMENSION KE1(30), KE2(30), KE3(30), KE4(30)
DIMENSION STRM(60,20), STRMC(60,10)
DIMENSION STRMI(8,20), STRMCI(8,10), STRMU(8,20), STR
IMCO(8,10)
DIMENSION DELS(20), DELSC(10)
DIMENSION TITLE(12), ICONV(15), NECALL(30)
REAL K,LHV

```

C  
C  
C  
C  
C  
C  
C

```

SUBROUTINE EQUILK

```

```

LATENT HEAT OF VAPOURIZATION IS ASSUMED CONSTANT FOR
EVALUATION OF EQUILIBRIUM CONSTANTS
IDEAL EQUILIBRIUM CONSTANTS USED

```

```

LHV=CMPROP(I,1)
VP=CMPROP(I,7)*EXP( LHV *CMPROP(I,8)/1.987*(1./T
IREF-1./T))
K=VP/P
RETURN
END

```



```

SUBROUTINE QUAL(N)
COMMON  KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON  KEFLAG, NE, NEMAX
COMMON  KSFLAG, NS, NSMAX
COMMON  EN, NFN, NEL, NELMAX,          ENC, NENC, NECL, NE
1CLMX
COMMON  SN, NSN, NSL, NSLMAX,          SNC, NSNC, NSCL, NS
1CLMX
COMMON  KLS, NUS, NUSMAX,              KPS, NPS, NPSMAX
COMMON  NLS1, NLS2, NUS3,              KCV1, KCV2, KCV3
COMMON  KEUS, NEUS, NENUM
COMMON  KECV1, KECV2, KECV3, KECV4
COMMON  NECV1, NECV2, NECV3, NECV4
COMMON  KE1, NE1, NE1MAX,              KE2, NE2, NE2MAX
COMMON  KE3, NE3, NE3MAX,              KE4, NE4, NE4MAX
COMMON  KES, KES2, KES3
COMMON  KRET, KRET2, KRET3
COMMON  STRM, STRMC
COMMON  STRMI, STRMCI, NIN,            STRMO, STRMCO, NOUT
COMMON  LELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON  KRUN, KSETS, KCLEAN, LOOPS, NOGO
COMMON  TITLE, NOKPM, NCEN, NOENC, NCSN, NOSNC, NCKPS,
1 NSTRM
COMMON  NLX, NLY, NLXS, NLXSC
COMMON  ICONV, NEX, NECALL, NEQUIP
COMMON  ND2, ND3, ND4, ND5, ND6, ND7
COMMON  EXX, D15, D16
COMMON  I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I1
15
COMMON  J2, J3, J4, J5
COMMON  J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17, J
118, J19, J20,
1 J21, J22, J23, J24, J25, J26, J27, J28, J29, J30, J31, J3
12, J33, J34,
2 J35, J36, J37, J38, J39, J40, J41, J42, J43, J44, J45, J4
16
COMMON  J47, J48, J49, J50, J51, J52, J53, J54, J55, J56, J57, J5
18, J59, J60
COMMON  K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, K1
15, K16, K17,
1 K18, K19, K20, K21, K22, K23, K24, K25, K26, K27, K28, K2
19, K30
COMMON  L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13
COMMON  NDELS, NDELSC
COMMON  NEXEQN, NAME, KSEM
COMMON  NIPT(30), NOPT(30), LIP, LOP, NOCOMP, NUNIT(30), LUNI
1T, NB, NONCAL,
1NALLCW(15), NLEGTN, NSTR(30), COEFF(16,16), C(16), NKSTR(16
1), NKCOL(16),
2NROW, NXB1, CMPRCP(15,10), NEB, TREF
COMMON  NEXEQN(30), NAME(30), KSEM(60,3)
COMMON  KPM(30,10), KEFLAG(30), KSFLAG(60), KPMR(30
1,12)
COMMON  SN(60,20), SNC(60,10), EN(30,10), ENC(30,10
1)

```





```

DIMENSION KUS(60), KPS(10), KEUS(8)
DIMENSION KE1(30), KE2(30), KE3(30), KE4(30)
DIMENSION STRM(60,20), STRMC(60,10)
DIMENSION STRMI(8,20), STRMCI(8,10), STRMD(8,20), STR
1MCC(8,10)
DIMENSION DELS(20), DELSC(10)
DIMENSION TITLE(12), ICONV(15), NECALL(30)
DIMENSION Z(8)
REAL K

```

```

C
C SUBROUTINE QUAL
C
C N IS THE STREAM NUMBER
C IDEAL K VALUES USED
C CLAUSIUS CLAPYRON EQUATION USED TO CALCULATE VAPOUR
C PRESSURES
C

```

```

NSCRPT=2*NCCOMP+4
T=SN(N,NSCRPT)
P=SNC(N,3)
IF(T.LT.SNC(N,6))GOTO3
IF(T.GT.SNC(N,7))GOTO4

```

```

C
C FLASH CALCULATION REQUIRED
C

```

```

GOTO5
3 SNC(N,5)=0.
RETURN
4 SNC(N,5)=1.
RETURN
5 VC=0.
SUMMS=0.
SUMZM=0.
DO 11 I=1,NCCOMP
11 SUMZM=SUMZM+SN(N,I+3)/CMPROP(I,8)
DO 12 I=1,NCCOMP
Z(1)=SN(N,I+3)/CMPROP(I,8)/SUMZM
12 SUMMS=SUMMS+Z(1)*CMPROP(I,8)
7 SUM=0.
SUMP=0.
SUMMG=0.
DO 6 I=1,NCCOMP
CALL EQUILK(I,K,T,P)
YG=K*Z(I)/(VO*(K-1.)+1.)
SUM=SUM+YG
SUMMG=SUMMG+YG*CMPROP(I,8)
6 SUMP=SUMP-K*Z(I)*(K-1.)/(VO*(K-1.)+1.）**2
SUM=SUM-1.
V=VO-SUM/SUMP
IF(ABS(V-VO).LT.0.001)GOTO8
10 VO=V
GOTO7
8 IF(V.LT.0.)V=0.
IF(V.GT.1.)V=1.
SNC(N,8)=V

```





```
SNC(N,5)=V*SUMMG/SUMMS  
RETURN  
END
```



```

SUBROUTINE BUBPT(N)
COMMON KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON KEFLAG, NE, NEMAX
COMMON KSFLAG, NS, NSMAX
COMMON EN, NEN, NEL, NELMAX, ENC, NENC, NECL, NE
1CLMX
COMMON SN, NSN, NSL, NSLMAX, SNC, NSNC, NSCL, NS
1CLMX
COMMON KUS, NUS, NUSMAX, KPS, NPS, NPSMAX
COMMON NUS1, NUS2, NUS3, KCV1, KCV2, KCV3
COMMON KEUS, NEUS, NENUM
COMMON KECV1, KECV2, KECV3, KECV4
COMMON NECV1, NECV2, NECV3, NECV4
COMMON KE1, NE1, NELMAX, KE2, NE2, NE2MAX
COMMON KE3, NE3, NE3MAX, KE4, NE4, NE4MAX
COMMON KES, KES2, KES3
COMMON KRET, KRET2, KRET3
COMMON STRM, STRMC
COMMON STRMI, STRMCI, NIN, STRMO, STRMCO, NOUT
COMMON DELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON KRLN, KSETS, KCLEAN, LOOPS, NOGO
COMMON TITLE, NOKPM, NOEN, NOENC, NCSN, NOSNC, NOKPS,
1 NSTRM
COMMON NLX, NLY, NLXS, NLXSC
COMMON ICONV, NEX, NECALL, NEQUIP
COMMON ND2, ND3, ND4, ND5, ND6, ND7
COMMON EXX, D15, D16
COMMON I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I1
15
COMMON J2, J3, J4, J5
COMMON J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17, J
118, J19, J20,
1 J21, J22, J23, J24, J25, J26, J27, J28, J29, J30, J31, J3
12, J32, J34,
2 J35, J36, J37, J38, J39, J40, J41, J42, J43, J44, J45, J4
16
COMMON J47, J48, J49, J50, J51, J52, J53, J54, J55, J56, J57, J5
18, J59, J60
COMMON K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, K1
15, K16, K17,
1 K18, K19, K20, K21, K22, K23, K24, K25, K26, K27, K28, K2
19, K30
COMMON L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13
COMMON NDELS, NDELSC
COMMON NEXEEN, NAME, KSEM
COMMON NIPT(30), NOPT(30), LIP, LOP, NCCOMP, NUNIT(30), LUNI
1T, NB, NNOAL,
1NALLOW(15), NLEGTB, NSTR(30), COEFF(16,16), C(16), NKSTR(16
1), NKCOL(16),
2NROW, NXB1, CMPROCP(15,10), NEB, TREF
DIMENSION NEXEEN(30), NAME(30), KSEM(60,3)
DIMENSION KPM(30,10), KEFLAG(30), KSFLAG(60), KPMR(30
1,12)
DIMENSION SN(60,20), SNC(60,10), EN(30,10), ENC(30,10
1)

```







```

DIMENSION KUS(60), KPS(10), KEUS(8)
DIMENSION KE1(30), KE2(30), KE3(30), KE4(30)
DIMENSION STRM(60,20), STRMC(60,10)
DIMENSION STRMI(8,20), STRMCI(8,10), STRMD(8,20), STR
1MCC(8,10)
DIMENSION DELLS(20), DELSC(10)
DIMENSION TITLE(12), ICONV(15), NECALL(30)
DIMENSION X(8)
REAL K

```

```

C
C SUBROUTINE BURPT
C
C SNC(NSN,6) IS THE BUBBLE POINT TEMPERATURE IN DEGREES
C RANKINE
C IDEAL K VALUES USED
C
IF(SNC(N,4).LT.1.9)GOTO9
T=CMPROP(1,2)
P=SNC(N,3)
KTIMES=1
C CONVERT MASS FRACTIONS TO MOLE FRACTIONS
SUMXM=C.
DO 13 I=1,NCCOMP
13 SUMXM=SUMXM+SN(N,I+3)/CMPROP(I,8)
DO 14 I=1,NCCOMP
14 X(I)=SN(N,I+3)/CMPROP(I,8)/SUMXM
3 SUMY=0.
DO 4 I=1,NCCOMP
CALL EQUILK(I,K,T,P)
Y=K*X(I)
4 SUMY=SUMY+Y
IF(ABS(SUMY-1.0)-0.01)8,8,5
5 KTIMES=KTIMES+1
IF(KTIMES)7,6,6
6 SUMYC=SUMY
TO=T
T=T+10.
GOTO3
7 SLOPE=(SUMY-SUMYC)/(T-TO)
TN=((1.0-SUMY)/SLOPE)+T
SUMYC=SUMY
TO=T
T=TN
GOTO3
8 SNC(N,6)=T
RETURN
9 SNC(N,6)=1./((1./TREF-2.303*10.73/CMPROP(1,8)*ALOG10(SN
1C(N,3)/CMPPC
1P(1,7))/CMPROP(1,1))
SNC(N,7)=SNC(N,6)
RETURN
END

```



```

SUBROUTINE DEWPT(N)
COMMON KPM, N2, N3, N2MAX, N3MAX, KPMR
COMMON KEFLAG, NE, NEMAX
COMMON KSFLAG, NS, NSMAX
COMMON EN, NEN, NEL, NELMAX, ENC, NENC, NECL, NE
1CLMX
COMMON SN, NSN, NSL, NSLMAX, SNC, NSNC, NSCL, NS
1CLMX
COMMON KLS, NUS, NUSMAX, KPS, NPS, NPSMAX
COMMON NLS1, NLS2, NUS3, KCV1, KCV2, KCV3
COMMON KEUS, NEUS, NENUM
COMMON KECV1, KECV2, KECV3, KECV4
COMMON NECV1, NECV2, NECV3, NECV4
COMMON KE1, NE1, NE1MAX, KE2, NE2, NE2MAX
COMMON KE3, NE3, NE3MAX, KE4, NE4, NE4MAX
COMMON KES, KES2, KES3
COMMON KRET, KRET2, KRET3
COMMON STRM, STRMC
COMMON STRMI, STRMCI, NIN, STRMO, STRMCO, NOUT
COMMON DELS, DELSC, LIMIT, LIMIT2, LIMIT3, LOOP, KBAC
1K
COMMON KRUN, KSETS, KCLEAN, LOOPS, NOGO
COMMON TITLE, NCKPM, NOEN, NOENC, NCSN, NOSNC, NOKPS,
1 NSTRM
COMMON NLX, NLY, NLXS, NLXSC
COMMON ICCNV, NEX, NECALL, NEQUIP
COMMON ND2, ND3, ND4, ND5, ND6, ND7
COMMON EXX, D15, D16
COMMON I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I1
15
COMMON J2, J3, J4, J5
COMMON J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17, J
118, J19, J20,
1 J21, J22, J23, J24, J25, J26, J27, J28, J29, J30, J31, J3
12, J33, J34,
2 J35, J36, J37, J38, J39, J40, J41, J42, J43, J44, J45, J4
16
COMMON J47, J48, J49, J50, J51, J52, J53, J54, J55, J56, J57, J5
18, J59, J60
COMMON K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, K1
15, K16, K17,
1 K18, K19, K20, K21, K22, K23, K24, K25, K26, K27, K28, K2
19, K30
COMMON L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13
COMMON NDELS, NDELSC
COMMON NEXEQN, NAME, KSEM
COMMON NIPT(30), NOPT(30), LIP, LOP, NCCOMP, NUNIT(30), LUNI
1T, NB, NONCAL,
1NALLCW(15), NLEGTN, NSTR(30), COEFF(16, 16), C(16), NKSTR(16
1), NKCOL(16),
2NRCH, NXB1, CMPPCP(15, 10), NEB, TREF
DIMENSION NEXEQN(30), NAME(30), KSEM(60, 3)
DIMENSION KPM(30, 10), KEFLAG(30), KSFLAG(60), KPMR(30
1, 12)
DIMENSION SN(60, 20), SNC(60, 10), EN(30, 10), ENC(30, 10
1)

```





```

DIMENSION KUS(60), KPS(10), KEUS(8)
DIMENSION KE1(30), KE2(30), KE3(30), KE4(30)
DIMENSION STRM(60,20), STRMC(60,10)
DIMENSION STRMI(8,20), STRMCI(8,10), STRMO(8,20), STR
IMCO(8,10)
DIMENSION DELS(20), DELSC(10)
DIMENSION TITLE(12), ICONV(15), NECALL(30)
DIMENSION Y(8)
REAL K

```

```

C
C SUBROUTINE DEWPT
C
C SNC(NSN,7) IS THE DEW POINT TEMPERATURE IN DEGREES
C RANKINE
C IDEAL K VALUES USED
C

```

```

T=CMFPROP(1,2)
P=SNC(N,3)
KTIMES = 1
C CONVERT MASS FRACTIONS TO MOLE FRACTIONS
SUMYM=C.
DO 13 I=1,NCCOMP
13 SUMYM=SUMYM+SN(N,I+3)/CMFPROP(1,8)
DO 14 I=1,NCCOMP
14 Y(I)=SN(N,I+3)/CMFPROP(1,8)/SUMYM
3 SUMX=C.
DO 4 I=1,NCCOMP
CALL EQUILK(I,K,T,P)
X=Y(I)/K
4 SUMX=SUMX+X
IF(ABS(SUMX-1.)-0.01)8,8,5
5 KTIMES=KTIMES+1
IF(KTIMES)7,6,6
6 SUMXC=SUMX
TC=T
T=T+10.
GOTO3
7 SLOPE=(SUMX-SUMXC)/(T-TC)
TN=((1.0-SUMX)/SLOPE)+T
SUMXC=SUMX
TC=T
T=TN
GOTO3
8 SNC(N,7)=T
RETURN
END

```











**B29864**